
TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2646 – Elektrotechnika a informatika

Studijní obor: 1802R007 – Informační technologie

Tiskový výstup webových aplikací

Print output from the web application

Bakalářská práce

Autor: **Lukáš Honc**

Vedoucí práce: Ing. Mojmír Volf

V Liberci 17. 5. 2013

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Lukáš Honc**
Osobní číslo: **M09000136**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **Tiskový výstup z webových aplikací**
Zadávací katedra: **Ústav nových technologií a aplikované informatiky**

Z á s a d y p r o v y p r a c o v á n í :

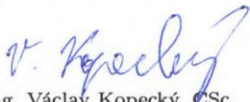
1. Proveďte rešerši dostupných technologií pro tiskové výstupy z webových aplikací.
2. Analyzujte použitelné technologie z výše popsaných hledisek.
3. Navrhněte a implementujte klientský nebo serverový skript pro tiskový výstup.
4. Vytvořte ucelenou technickou dokumentaci pro praktické nasazení skriptu.

Rozsah grafických prací: **dle potřeby**
Rozsah pracovní zprávy: **45 stran**
Forma zpracování bakalářské práce: **tištěná/elektronická**
Seznam odborné literatury:

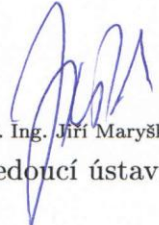
- [1] VRÁNA, Jakub. 1001 Tipů a triků pro PHP, Brno, Computer Press, 2012.
- [2] Leiss Oliver, Schmidt Jasmin, PHP v praxi, Grada, 2010.
- [3] Öggl Bernd, Kofler Michael, PHP 5 a MySQL 5, Computer Press, 18.10.2007.
- [4] Odell Den, Javascript, Computer press, 27.4.2010.
- [5] JQuery dokumentace, 2010. Dostupné z: <http://docs.jquery.com/>
- [6] Dokumentace k CSS, 1994-2012, W3C member organizations. Dostupné z: <http://www.w3.org/Style/CSS/>
- [7] Škola pro webmastery, 1999 - 2012, W3Schools. Dostupnost z: <http://www.w3schools.com>
- [8] MySQL manuál, 2012, Oracle Corporation. Dostupné z: <http://dev.mysql.com/doc/>
- [9] PHP dokumentace, 2001 - 2012, The PHP group. Dostupné z: <http://php.net/>
- [10] PHP triky, 2005 - 2012, Jakub Vrána. Dostupné z: <http://php.vrana.cz>

Vedoucí bakalářské práce: **Ing. Mojmír Volf**
Ústav nových technologií a aplikované informatiky

Datum zadání bakalářské práce: **18. října 2012**
Termín odevzdání bakalářské práce: **17. května 2013**


prof. Ing. Václav Kopecký, CSc.
děkan




prof. Dr. Ing. Jiří Maryška, CSc.
vedoucí ústavu

V Liberci dne 18. října 2012

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum: 16.5.2013

Podpis

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Mojmíru Volfovi za odborné vedení této bakalářské práce a za cenné připomínky a rady v průběhu jejího řešení.

ANOTACE

Tato bakalářská práce se zabývá tiskovým výstupem webových aplikací. Jejím cílem bylo porovnat a vyhodnotit jednotlivé tiskové výstupy těchto aplikací, a následně vyhodnotit export do souboru formátu PDF. Tiskový výstup webových aplikací je hojně rozšířená metoda využití informací z internetu. Pro export webových aplikací do formátu PDF byla v bakalářské práci použita třída MPDF, která je pro tento export přímo vytvořena. Analýzou exportů této třídy byly zjištěny její nedostatky a rozdíly zobrazení vyexportovaného souboru s náhledem v internetovém prohlížeči, zejména Google Chrome a Mozilla Firefox. Bakalářská práce se zabývá podrobnou funkcionalitou této třídy a její porovnání s několika konkurenčními softwary, zejména třídami FPDF a HTML2(F)PDF.

Klíčová slova

Tiskový výstup, MPDF třída, export do PDF souboru, tisk z internetového prohlížeče

ANNOTATION

This bachelor thesis deals with print output of web applications. The purpose of this thesis is to compare and evaluate individual print outputs of these applications and then to evaluate an export to PDF file. The print output of web applications is a widely spread method of using information from the Internet. For the export of web applications to PDF file there was used the MPDF class, which is created directly for this export. By the analysis of exports of this class there were found its deficiencies and differences of view of an exported file with a preview in web browsers, mainly in Google Chrome and Mozilla Firefox. This bachelor thesis follows up detailed functionality of this class and its comparison with several competitive softwares, especially with classes FPDF and HTML2(F)PDF.

Keywords

Print output, MPDF class, export to PDF file, print from the web browser

Obsah

ANOTACE	6
1. ÚVOD	9
2. TISKOVÉ VÝSTUPY WEBOVÝCH APLIKACÍ	11
2.1. KASKÁDOVÉ STYL	11
2.1.1. <i>Meziplatformní kompatibilita</i>	12
2.1.2. <i>Technická náročnost</i>	13
2.1.1. <i>Renderovací jádro WebKit</i>	13
2.1.2. <i>Renderovací jádro Gecko</i>	13
2.1.3. <i>Renderovací jádro MSHTML</i>	14
2.1.4. <i>Rozdíl jader</i>	14
2.1.5. <i>Vhodnosti použití</i>	15
2.2. JAVASCRIPT	15
2.2.1. <i>Technická náročnost</i>	16
2.2.2. <i>Meziplatformní kompatibilita</i>	16
2.2.3. <i>Vhodnosti pro konkrétní oblast použití</i>	16
2.3. JQUERY	17
2.3.1. <i>Technická náročnost</i>	17
2.3.2. <i>Meziplatformní kompatibilita</i>	17
2.3.3. <i>Vhodnost pro konkrétní oblasti použití</i>	18
3. EXPORT WEBOVÝCH APLIKACÍ DO PDF SOUBORU	18
3.1. EXPORT DAT Z DATABÁZE MYSQL	18
3.1.1. <i>Meziplatformní kompatibilita</i>	19
3.1.2. <i>Technická náročnost</i>	19
3.1.3. <i>Vhodnost pro konkrétní oblasti použití</i>	20
3.2. FPDF	20
3.2.1. <i>Technická náročnost</i>	21
3.2.2. <i>Meziplatformní kompatibilita</i>	21
3.2.3. <i>Vhodnost pro konkrétní oblasti použití</i>	22
4. HTML2(F)PDF	23
4.1. TECHNICKÁ NÁROČNOST	23
4.2. MEZIPLATFORMNÍ KOMPATIBILITA	24
4.3. VHODNOST PRO KONKRÉTNÍ OBLASTI POUŽITÍ	24
5. MPDF	24
5.1. TECHNICKÁ NÁROČNOST	25
5.1.1. <i>Měření exportu tabulek pomocí MPDF</i>	25

5.1.2. Porovnání velikosti exportovaného souboru z MPDF a FPDF.....	27
5.2. MEZIPLATFORMNÍ KOMPATIBILITA	28
5.3. ROZDÍLNÉ ZOBRAZENÍ MEZI EXPORTEM POMOCÍ MPDF A V PROHLÍŽEČÍCH	30
5.3.1. Formuláře pomocí MPDF.....	30
5.3.2. Jednotky a rozdíl velikostí.....	32
5.4. HTML + CSS.....	32
5.4.1. Druhy zápisu HTML/CSS	33
5.4.2. Podpora HTML.....	34
5.5. FONTY, CIZÍ ZNAKY A ZRYCHLENÍ EXPORTU	36
5.6. EXPORT BLOKOVÝCH ELEMENTŮ	38
5.7. JAVASCRIPT	39
5.8. BARVY, OBRÁZKY A VEKTOROVÁ GRAFIKA	39
5.9. OVLÁDACÍ ZNAČKY TŘÍDY MPDF	40
5.9.1. Anotace	41
5.9.2. EAN Čárové kódy.....	41
5.9.3. Záložky	42
5.9.4. Sloupce.....	42
5.9.5. Tečkování.....	43
5.9.6. Ukončení stránky	43
5.9.7. Záhloví a zápatí stránky	44
5.9.8. Vykreslení textu do kruhu.....	44
5.9.9. Obsah	45
5.10. ROZŠÍŘENÍ PRO MPDF TŘÍDU A JEJÍ VÝVOJ	45
5.10.1. Grafy	45
5.10.2. QR kódy	46
5.10.3. Vývoj třídy.....	47
5.11. MPDF VS. FPDF S ROZŠÍŘENÍMI	49
6. ZÁVĚR.....	52
7. SEZNAM POUŽITÉ LITERATURY.....	54
8. PŘÍLOHY	56

1. Úvod

V dnešní době je nejrozšířenějším zdrojem informací internet, kde je umístěno nespočetné množství webových prezentací a aplikací. Tyto aplikace mohou obsahovat text, tabulky, videa, obrázky a další prostředky pro sdělení informací. Uživatelé využívající internet tyto informace prochází ve webovém prohlížeči, který mají nainstalován na svém zařízení. Typicky se jedná o stolní počítač, notebook, netbook, tablet a mnoho dalších elektronických zařízení, které umožňují surfování na internetu.

I když jsou počítače a obecně veškerá elektronika rozšířeny a hojně používány, v některých situacích se uživatelé bez vytisknutých informací z internetu neobejdou. K tomu je potřeba vytvořit tiskový výstup z webových aplikací, ke kterému je možné využít funkcí prohlížeče, který umožňuje tisk zobrazené stránky. Prohlížeče ale umožňují pouze náhled na stránku a následný tisk, nikoliv však export těchto dat do souboru jiného formátu než HTML.

Při exportu webových aplikací do souboru je nejčastěji využíván export do formátu PDF, který vyvinula firma Adobe. Tento soubor se na všech platformách zobrazuje totožně, a k jeho zobrazení stačí použít prohlížeče k tomu určené. Nejznámějším prohlížečem PDF souborů je software vyvinut firmou Adobe - Adobe Reader. Ukládání dat do PDF souboru je nezávislé na hardwaru i softwaru stanice, na které byl soubor vytvořen.

Pro export webových aplikací do formátu PDF se využívají klientské nebo serverové skripty. Nejčastěji používaný klientský programovací jazyk je JavaScript. Klientské skripty pracují na straně klienta, v tomto případě v internetovém prohlížeči. Tyto skripty obsahují funkce pro ovládání prohlížeče, který reaguje dle požadavků uživatele bez komunikace se serverem. Skripty jsou závislé na uživatelské platformě, konkrétně tedy na podporovaných funkcích prohlížeče. Z tohoto důvodu nemusí klientské skripty fungovat ve všech prohlížečích totožně.

Serverové skripty pracují na serveru, a nejsou tedy závislé na uživatelské platformě. Zpravidla se jedná o jazyk PHP. Tyto skripty jsou schopny používat pouze data na serveru, ne data u uživatele. Pro export do souboru formátu PDF je možné využít buď předkompilovanou třídu PHP – PDFlib, která nepodporuje zápis HTML ani CSS. Je také možné využít třídy, které jsou určeny pro převod jazyka HTML

do souboru PDF. Těchto tříd existuje několik, ale každá má několik omezení. Je možné použít třídy s podporou nebo bez podpory zápisu kaskádových stylů, které umožňují grafické formátování a stylování webové aplikace.

Pro export webových aplikací do formátu PDF byla vybrána třída MPDF, která byla vytvořena přímo pro tento druh exportu. Třída podporuje UTF-8 kódování, české znaky s diakritikou, dokonce i asijské, indické a egyptské jazyky a jejich značky. Tato knihovna je vydávána pod GPL licencí. Je tedy možné ji volně používat a upravovat. Exportované stránky jsou ve výsledném PDF souboru zobrazeny téměř shodně, jako v internetovém prohlížeči. Jelikož se jedná o převod jazyka HTML do formátu PDF, je nutné tento jazyk konvertovat, aby se v PDF souboru zobrazoval shodně, jako v internetovém prohlížeči.

2. Tiskové výstupy webových aplikací

Pro vytisknutí webové aplikace z prohlížeče je možné využít funkci internetového prohlížeče. Funkce pro tisknutí umožňuje pouze odeslání dat do tiskárny, nikoliv export stránky do konkrétního formátu souboru. Tisknutí z webového prohlížeče lze uživatelům zjednodušit pomocí klientského skriptu, například JavaScript, který tuto funkci udělá. Je však nutné, aby uživatel tuto funkci zavolał. Pro export dat z databáze do souboru je možné využít serverový skriptovací jazyk, například PHP. V tomto jazyce byly vytvořeny přímo aplikace pro ovládání databáze, které umožňují export dat do různých formátů souboru. Jedná se například o aplikaci PHPMyAdmin, nebo Adminer. Je možné exportovat záznamy z tabulek, popřípadě záznamy celé databáze do PDF souboru.

2.1. Kaskádové styly

CSS je jazyk určený pro způsob zobrazení stránek v internetovém prohlížeči. Stránky jsou psány v jazycích HTML a XML. Cílem kaskádových stylů je oddělit informace obsahu stránky od formátovacích pokynů. Informativní část stránek je tvořena jazykem HTML, zatímco kaskádové styly slouží pro formátování obsahu stránky. Pravidla pro zápis kaskádových stylů a správné zobrazení v prohlížečích určuje W3C. W3C je společnost specifikující standardy WWW (World Wide Web) pro dlouhodobý růst webových aplikací.

Internetový prohlížeč umožňuje uživatelům tisk webových aplikací, které jsou aktuálně načteny v internetovém prohlížeči. V Menu prohlížeče je možnost Soubor -> Tisk. Po provedení příkazu je v prohlížeči zobrazen náhled stránky pro tisk. Po potvrzení tisku je aktuálně načtená stránka v prohlížeči odeslána do příslušné tiskárny. U prohlížeče Internet Explorer je náhled přeskočen a je ihned zobrazena možnost volby tiskárny a parametrů pro tisk. Pro náhled tisku v tomto prohlížeči je potřeba v menu vybrat položku Náhled tisku.

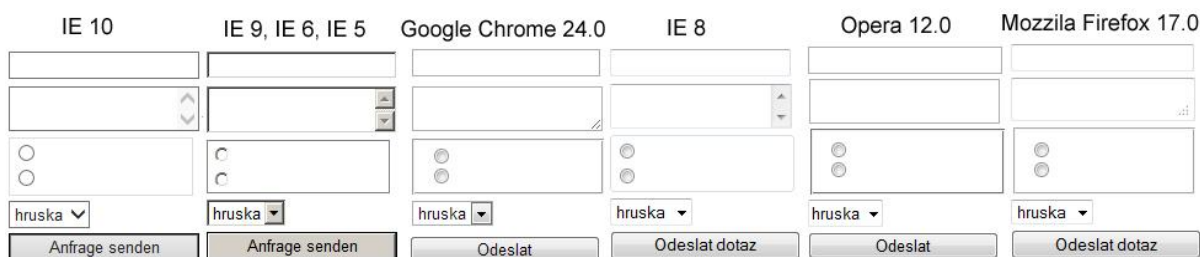
V aktuálně načtených webových aplikacích v prohlížeči mohou být zobrazeny prvky, které není třeba tisknout, nebo je potřeba změnit jejich barva písma a pozadí pro lepší viditelnost při tisku. Proto vznikla možnost definovat kaskádový styl, který je určen pro tiskový výstup. Tento soubor se importuje do stránky značkou(tagem) `<link>`

a je popsán pomocí atributu *media*, který musí být definován pro tisk, tedy nabývat hodnoty *print*. Tato varianta je zahrnuta i v hodnotě *all* tohoto atributu, tedy pro všechna zobrazení – jak v prohlížeči, tak při náhledu k tisku. Při tvorbě nového CSS souboru je doporučováno definovat výchozí font pro písmo a reset nastavených pozic všech prvků aplikace. Font písma by měl být dobře čitelný v tištěné podobě, doporučeno je používat Arial, Helvetica a jiná bezpatková písma. V zásadě je nejlepší použít 5 základních fontů – Arial, Courier, Times New Roman, Tahoma, Verdana, které by měly být součástí všech výchozích instalací webových prohlížečů. Reset pozic prvků způsobí, že se všechny zobrazí bez nastavené konkrétní pozice. Je vhodné tyto pozice uzpůsobit tisku, nejčastěji rozměru stránky A4, a ne rozměru okna prohlížeče

Prvky, které se zpravidla netisknou, jsou formuláře, postranní panel (SideBar), hlavička stránky(Header), social bookmarking, atd. Tyto prvky je doporučeno při definování stylu pro tisk skrýt. Stejně tak nepoužívat obrázky na pozadí, nebo špatný kontrastní poměr mezi barvou písma a barvou pozadí. Při tisku také nejsou vidět URL adresy, na které směřují odkazy. Je vhodné je pomocí kaskádových stylů při tisku zviditelnit.

2.1.1. Meziplatformní kompatibilita

Jazyk CSS není závislý na platformě. Jeho chování by se nemělo měnit v závislosti na operačním systému. Ale každý operační systém může mít některé části definovány jinak, nebo je nemusí vůbec podporovat. Například některý ze základních fontů od Microsoftu nemusí být implementován ve výchozí instalaci Linuxu. Zobrazení pomocí kaskádových stylů je závislé na prohlížeči, protože každý prohlížeč může interpretovat vlastnosti jinak, nebo je nemusí vůbec podporovat. Za předpokladu shodné interpretace kaskádových stylů je v prohlížečích stejně vidět rozdílné zobrazení u formulářů. Rozdíl je vidět především u internetových prohlížečů Google Chrome a Opera, protože tyto prohlížeče mají definovány své vlastní inputy.



Obrázek 1 - Rozdílné zobrazení formulářů v internetových prohlížečích

Problém při čitelnosti tiskového výstupu se může vyskytnout i v případě, že uživatel si ve svém prohlížeči nadefinuje jiné výchozí barvy pozadí a písma. Proto je doporučováno při tisku tyto barvy definovat, aby nedošlo k nečitelnému tisku, nejčastěji barvu písma na černou a barvu pozadí na bílou barvu.

2.1.2. Technická náročnost

Soubor s kaskádovým stylem určený pro tisk se odesílá klientovi při načítání webové aplikace. Jazyky HTML ani CSS nejsou určené k obsluze tisku, a tuto funkci neumí. Tisk webových aplikací, napsaných v těchto jazycích, musí obsloužit jiný programovací jazyk, popřípadě webový prohlížeč. Součástí prohlížeče je vždy renderovací jádro, které obsluhuje zobrazení a vykreslení webových aplikací u klienta. Jader je několik druhů. Nejrozšířenější je jádro WebKit, které používají internetové prohlížeče Google Chrome a Apple Safari. Mozilla Firefox používá pro zobrazování webových aplikací jádro Gecko a Internet Explorer jádro MSHTML.

2.1.1. Renderovací jádro WebKit

Jádro WebKit pro vykreslení webových aplikací využívá z nejrozšířenějších internetových prohlížečů Google Chrome a Safari. Google Chrome však do budoucna nebude založen na tomto jádru, ale vytvoří si vlastní jádro – Blink, které vychází z jádra WebKit. Webové aplikace, které jsou zobrazovány v prohlížečích založených na tomto jádře, jsou nejdříve parsovány pomocí HTML parseru, který z jazyka HTML na načítané stránce vytvoří stromovou strukturu s DOM uzly, tzv. DOM strom. Zároveň pomocí CSS parseru analyzuje vlastnosti připojených kaskádových stylů, z kterých jsou vytvořena pravidla stylů. Spojením DOM stromu a pravidel kaskádových stylů je vytvořen „Render tree“, který je nejdříve u uživatele postupně vykreslován pomocí konkrétních souřadnic definovaných buď v HTML, nebo v kaskádovém stylu, a poté zobrazen jako načtená stránka.

2.1.2. Renderovací jádro Gecko

Gecko využívá internetový prohlížeč Mozilla Firefox. Gecko v prvním kroku parsuje HTML kód celé stránky pomocí HTML parseru, z kterého vytvoří „Content Sink“, ve kterém jsou uloženy DOM elementy. Jedná se o mezivrstvu mezi HTML a DOM stromem, kterou WebKit nemá. DOM element vždy obsahuje konkrétní značku

se všemi jeho atributy. V následujícím kroku vytvoří pomocí DOMu obsah modelu z „Content Sink“ a pomocí CSS parseru vytvoří pravidla stylů z připojených CSS souborů. Spojením obsahu modelu a pravidel stylů zkonstruuje „Frame Tree“, do kterého formátuje zobrazení prvků. Tento strom je u klienta nejdříve vykreslen s přesnými pozicemi a vlastnostmi jednotlivých prvků, a poté zobrazen jako aktuálně načtená stránka v prohlížeči.

2.1.3. Renderovací jádro MSHTML

Toto jádro je součástí každé instalace operačního systému Windows a je využíváno i jinými programy než internetovým prohlížečem Internet Explorer. Jádro MSHTML vyvinula firma Microsoft. Jádro se skládá z několika částí, a to Trident, což je jádro pro vykreslení a zobrazení webových aplikací, dále HTML a CSS parser pro parsování přijatých dat, DOM a DHTML pro ovládání prvků stránky, a ActiveDocument pro aktivní dokumenty a jejich rozšíření technologií OLE.

Jádro MSHTML používá 3 základní struktury pro vykreslení webových aplikací v klientovi. Jsou to struktury:

- HTML_PAINT_DRAW_INFO, která poskytuje informace o vykreslovaném chování prvků.
- HTML_PAINT_XFORM, která představuje matici prvků
- HTML_PAINTER_INFO a nese informace o překládání a transformaci jednotlivých prvků matice, a
- HTML_PAINTER_INFO, která zpracovává informace o chování, funkčnosti a způsobu vykreslení jednotlivých prvků.

2.1.4. Rozdíl jader

Gecko vytváří tzv. „Frame Tree“, kde je každý prvek reprezentován jako rám, zatímco WebKit používá tzv. „Render tree“ z „render objektů“. Gecko má mezivrstvu mezi HTML parserem a DOMem, která vytváří DOM prvky – „Content Sink“. Renderovací jádro Trident bylo vyrobeno ve firmě Microsoft, která utajila jeho kód. Zveřejněna je pouze dokumentace k tomuto jádru.

2.1.5. Vhodnosti použití

Použití kaskádových stylů je doporučeno pro všechny internetové aplikace, u kterých chce programátor stránek definovat konkrétní design, a jsou psány v jazycích HTML a XML. Bohužel ne vždy všechny prohlížeče interpretují vlastnosti kaskádových stylů shodně. Je to způsobeno špatným pochopením, nebo implementací vlastností kaskádových stylů při vývoji prohlížeče. Pro správné zobrazení vlastností kaskádových stylů jsou definovány standardy společnosti W3C. Nejvíce chybných zobrazení obsahoval Internet Explorer 6, který interpretoval některé CSS vlastnosti chybně. Například box model, který slouží pro nastavení a konkrétní umístění blokových prvků na stránce. U tohoto modelu pracuje Internet Explorer 6 chybně s vlastnostmi *width* a *height*, tedy výškou a šířkou boxu. Dle standardů W3C, konkrétně standard 8 vlastností CSS 2, šířka a výška boxu je určena rozměry daného blokového prvku, například značka *div*. Šířka u Internet Exploreru 6 znamená šířku rámečku, který je od boxu odsazen pomocí vlastnosti *padding*.

Problém prohlížečů je, že ne vždy nová verze rozvíjí tu starou. Co se ve starších verzích zobrazovalo dle požadavků programátora webové aplikace, v nové verzi již nemusí. Typický příklad – vývoj Internet Exploreru, který postupně přecházel ze standardů firmy Microsoft na standardy W3C.

Pro správné zobrazení ve všech prohlížečích je možné použít tzv. CSS hacky. Jde o zápis CSS selektoru, při kterém dojde k porušení pravidel W3C, ale stránky se budou zobrazovat dle požadavků programátora ve všech prohlížečích. Každá verze prohlížeče rozumí jinému hacku. IE6 například hvězdičce nebo podtržítka před selektorem nebo vlastností v kaskádě. IE 7 a 8 zase „/***/“ za vlastností CSS a „\9“ za hodnotou vlastnosti, před středníkem.

2.2. JavaScript

JavaScript je klientský skriptovací jazyk. Skript napsaný v tomto jazyce funguje na straně klienta, v tomto případě ve webovém prohlížeči. Pomocí tohoto jazyka je možné tisknout webové stránky metodou `Print()`, která patří objektu `Window`. Objekt `Window` reprezentuje aktuálně načtenou stránku v internetovém prohlížeči. Po spuštění této události se zobrazí náhled tisku. Metoda `Print()` funguje na stejném principu jako

funkce internetového prohlížeče Soubor -> Tisk. Metoda tedy volá funkci internetového prohlížeče.

2.2.1. Technická náročnost

Po zavolání události si prohlížeč načte soubory kaskádových stylů určené pro tisk. Jsou to soubory, které obsahují atribut *media* s hodnotou *print*, nebo *all*. Prohlížeč pro konvertování na tiskový výstup použije HTML parser a CSS parser, díky kterým vytvoří DOM strom z HTML značek a CSS předpisy z kaskádových stylů. Spojením DOM stromu a CSS předpisů vznikne Render strom, který je zobrazen v prohlížeči při náhledu pro tisk. Pomocí JavaScriptu je možné tisknout stránky až po jejím plném načtení. Pro uživatele s pomalým internetovým připojením, nebo pro rozsáhlé stránky to může znamenat určité omezení. Pokud totiž dojde k přerušení načítání stránky, nemusí tisk pomocí JavaScriptu fungovat.

2.2.2. Meziplatformní kompatibilita

JavaScript je multiplatformní klientský skript, funguje tedy na různých platformách na straně klienta. Metoda Print() je podporována v nejrozšířenějších prohlížečích, tedy Internet Explorer, Mozilla Firefox, Google Chrome a Opera. Pomocí JavaScriptu je možné definovat události, které budou provedeny před tiskem - metoda `onBeforePrint()`, a po tisku – metoda `onAfterPrint()`. Metoda `onBeforePrint()` je možné využít pro informování uživatele o přípravě tisku. Zatímco metodu `onAfterPrint()` je vhodné využít pro informování o správně provedeném tisku, popřípadě pomocí jazyka Ajax informovat server o vytištění konkrétní stránky. To je vhodné pro majitele internetových stránek. Tyto události fungují od verze prohlížeče Internet Explorer 5 a novější, Firefox 6 a novější, Google Chrome 9 a novější a Safari 5.1 a novější. V prohlížeči Opera nejsou tyto události podporovány, tudíž nefungují. Z toho plyne, že skript napsaný v JavaScriptu, který správně funguje ve většině prohlížečů, nemusí ve všech prohlížečích fungovat shodně. Konkrétně funkcionalita skriptů závisí na jádru prohlížeče pro JavaScript.

2.2.3. Vhodnosti pro konkrétní oblast použití

JavaScript pro tisk je možné použít u všech stránek vytvořených v HTML. Tuto metodu je vhodné použít, pokud je stránka načtené webové aplikace rozdělena na

polovinu, například pomocí rámu, a je potřeba uživatelům umožnit vytisknout pouze jednu polovinu této aplikace. Metoda `Print()` musí být připojena k nějaké události, například událost `onClick` u tlačítka nebo u odkazu, protože jinak ji není možné zavolat.

Pomocí JavaScriptu je možné vytisknout internetovou stránku, která není aktuálně načtena v internetovém prohlížeči. Tento problém lze vyřešit buď pomocí rámu (značka *iframe*), ve kterém bude načtena požadovaná stránka, která má být odeslána k tisku, nebo druhou možností je webovou aplikaci načíst v nově otevřeném okně nebo kartě webového prohlížeče. Pro okamžité zavolání metody `Print()` po načtení aplikace je možné využít událost `onLoad`. Tato událost se nejčastěji připojuje k tělu dokumentu, tedy značce *body*, a je obsloužena ihned po načtení celé webové aplikace ve webovém prohlížeči.

2.3. JQuery

JQuery je JavaScriptový framework, umožňuje tedy ty samé funkce, jako JavaScript. K tisku je možné využít funkci `Print()`, která je použita i v JavaScriptu.

2.3.1. Technická náročnost

Jelikož se jedná o knihovnu, aplikace v ní napsané budou o něco pomalejší, než aplikace napsané přímo v jazyce JavaScript. JQuery má stejnou funkcionalitu jako JavaScript. Je možné tedy tisknout aktuálně načtené aplikace nebo stránky, které v prohlížeči nejsou načteny.

2.3.2. Meziplatformní kompatibilita

Při tvorbě JQuery knihovny si byli vývojáři vědomi rozdílného chování prohlížečů při spouštění událostí JavaScriptu. Proto hlavním cílem bylo, aby se všechny skripty psané v JQuery chovali v nejrozšířenějších prohlížečích totožně. Verze JQuery 1.X je podporována od internetového prohlížeče Internet Explorer 6.0, zatímco JQuery verze 2.X je podporována až od internetového prohlížeče Internet Explorer 9.0. Ve všech ostatních prohlížečích (Google Chrome, Mozilla Firefox a Opera) fungují obě verze. Pokud se v některém z prohlížečů chová jinak a skript je správně naprogramován, je dobré o tom informovat podporu pro JQuery.

2.3.3. Vhodnost pro konkrétní oblasti použití

Stejně jako Javascript i JQuery umožňuje tisk aktuální stránky, která je načtena v prohlížeči. Dále umožňuje tisk části stránky, která je načtena v rámu, nebo tisk stránky, která není v prohlížeči zobrazena, ale na kterou směřuje odkaz. JQuery je stále aktualizována a jsou vyvíjeny nové verze. Z tohoto důvodu je doporučováno knihovnu vždy stáhnout a nahrát na hosting ke konkrétní webové aplikaci, která ji používá, protože v nové verzi nemusí správně fungovat metody ze starších verzí. V horším případě nebudou fungovat vůbec. Hlavní výhodou JQuery je možnost jednoduchých změn v HTML a CSS, například změny pozadí dle uživatelských požadavků.

Javascript i JQuery mají jednu velkou nevýhodu – pokud uživatel nechce nebo nemůže používat JavaScript, jejich funkce nebudou fungovat. Proto je vhodné umístit na stránky prvky ovládající tisk klientským skriptem také pomocí tohoto skriptu. Pokud je totiž JavaScript vypnutý, prvky nejsou zobrazeny a aplikace funguje dle nastavení pro uživatele s tímto omezením.

3. Export webových aplikací do PDF souboru

Jazyk HTML nepodporuje export stránky do formátu PDF. Tento export není podporován ani v prohlížečích. Pro export webových aplikací do PDF je potřeba vytvořit skript, který bude tuto úlohu obsluhovat. K tomuto exportu je možné použít serverové nebo klientské skripty. Ze serverových skriptů sem patří například jazyk PHP, který zpracuje data na serveru, následně je odešle klientovi, a není tedy závislý na platformě uživatele. Z klientských skriptů to je například JavaScript, který pracuje s daty na straně klienta, je závislý na prohlížeči, ale nezatěžuje server. Pro oba druhy těchto programovacích jazyků bylo vytvořeno již několik knihoven, které jsou přímo určeny pro převod jazyka HTML do PDF formátu.

3.1. Export dat z databáze MySQL

Databáze neumožňují tiskový výstup. Databáze na serveru se nejčastěji ovládají přes administrační rozhraní, například phpMyAdmin. Toto rozhraní je naprogramováno v jazyce PHP, jedná se tedy o serverový skript. Při použití administračního rozhraní lze zavolat tiskový výstup buď pomocí internetového prohlížeče, nebo exportovat data do souboru. Pomocí internetového prohlížeče je ale k tisku odesláno vše, co je na stránce

zobrazeno. Mnohem vhodnější je využít export dat v tabulkách nebo celé databáze do různých formátů souboru. Pro tiskový výstup je nejvhodnější formát PDF, do kterého jsou data exportována z tabulky nebo z celé databáze. MySQL databázi lze spravovat pomocí phpMyAdminu, který pro export do PDF využívá FPDF knihovnu. Tato knihovna je pomalejší než předkompilovaná PDFlib knihovna, ale díky svým vlastnostem a jednoduššímu zápisu je upřednostňována před předkompilovanou knihovnou.

3.1.1. Meziplatformní kompatibilita

FPDF knihovna pracuje na PHP verzích 4 a 5 a nevyžaduje žádná rozšíření, pouze podporu komprese. Pro PHP 3 lze využít knihovnu FPDF do verze 1.4. Poslední verze FPDF vyžaduje pro funkčnost minimálně verzi PHP 4.3.10. Jelikož se jedná se o serverové řešení, není závislé na platformě uživatele.

Soubor PDF byl vyvinut tak, aby byl nezávislý na hardwaru i softwaru, ze kterého byl pořízen. Díky tomu jsou exportované soubory zobrazeny na všech zařízeních stejně. Soubor PDF umožňuje vytvářet dokumenty v mnoha jazycích, včetně mimo evropských.

3.1.2. Technická náročnost

Při exportu je nutné vybrat, z kterých tabulek je potřeba data exportovat, a definovat další parametry, jako znakovou sadu a kompresi souboru. U tabulek s větším obsahem záznamů je umožněno vybrat pouze některé z nich. Vždy je potřeba si zvolit, kolik řádků vytisknout a od kterého řádku začít, což může znamenat problém v případě, kdy je potřeba exportovat řádky, které nejdou po sobě. V takovémto případě je nutné přejít na stránku, kde se do databáze vkládají data s SQL dotazy, a dále pomocí příkazu *SELECT* vybrat záznamy, které mají být exportovány. Po potvrzení příkazu jsou vypsané záznamy, které splňují parametry příslušného dotazu. Na této stránce je umožněn export vybraných dat.

Po potvrzení exportu jsou nejprve vybrány názvy atributů a data z tabulky. Tyto data jsou pomocí PHP skriptu zapsána do proměnné, která je pomocí FPDF třídy vypsaná do PDF souboru ve formátu tabulky. V prvním řádku tabulky jsou názvy atributů, v dalších záznamy z dané entity. Nad tabulkou je název databáze a tabulky, z které byla data exportována. PDF soubor je po jeho dokončení a uzavření stažen ke

klientovi. Pokud si při exportu nedefinujeme jinak, název souboru je jméno databáze, z které byla data exportována.

Databáze: weap2, Tabulka: knizka

titul	zanr	kladne	zaporne	jmeno	prijmeni	nick
Kde domov	fantasy	26	4	Jaroslav	Seifert	
Ahojky	roman	1	1	Jaroslav	Seifert	
alkohol	roman	1	3	Jaroslav	Seifert	
seifert	roman	2	1	Jaroslav	Seifert	
Pokus1	roman	1	0	Arnold	Schwart	astek

Obrázek 2 - Exportovaná data tabulky z databáze do PDF

3.1.3. Vhodnost pro konkrétní oblasti použití

Export do formátu PDF z databáze je vhodný pro zobrazení obsahu tabulek nebo databáze v dobře čitelné podobě. Formát PDF je multiplatformní, lze ho tedy zobrazit na většině používaných platforem. Export dat z databáze používá pouze písmo Helvetica, které patří mezi 5 základních písem, které by měly být v každé výchozí instalaci operačního systému, a není tedy nutné tento font stahovat nebo zvětšovat výslednou velikost souboru připojením části nebo celého fontu k výslednému souboru.

Nevýhoda exportu dat z databáze do PDF souboru se vyskytne, pokud má tabulka mnoho atributů. Při tomto exportu je šířka tabulky přizpůsobena šířce stránce, a šířka jednotlivých atributů rovnoměrně rozdělena. Obsah jednotlivých entit může být zobrazen ve špatně čitelné podobě. V případě mnoho entit nemusí být jejich obsah zobrazen vůbec.

3.2. FPDF

FPDF je třída napsaná v jazyce PHP pod svobodnou licencí, která umožňuje export textu a jednoduchých grafických prvků do souboru PDF přímo z PHP, bez využití předkompilované knihovny PDFLib. Tato třída nepodporuje zápis HTML značek ani UTF-8 kódování. Pro podporu českých znaků s diakritikou je nutné vytvořit vlastní fonty pro PDF soubory s českou diakritikou. Podporuje obrázky pouze ve formátu JPG, PNG a GIF. Pro podporu GIF souborů je nutné mít nainstalované

rozšíření knihovnou GD. Pro podporu komprese exportovaných souborů je potřeba knihovna ZLIB. Třidu není nutné nijak instalovat, stačí ji pouze rozbalit v příslušném adresáři, kde bude exportovat dané dokumenty.

3.2.1. Technická náročnost

Serverové skripty mají přidělenou paměť, která je omezena pro daný skript. Zpravidla je to 8MB. Aby nedocházelo k vytížení serveru, je doba trvání skriptu nastavena na 30 sekund. Pokud dojde k překročení některého z těchto parametrů, je skript ukončen a částečně vytvořený soubor ztracen. U FPDF třídy lze předejít tomuto problému dynamickým nastavením doby běhu skriptu pomocí PHP funkce *set_time_limit()*, kde je možné změnit dobu trvání skriptu. Pokud toto nastavení bude nedostačující, je doporučeno vytvářet velké dokumenty do souboru na serveru a odesílat data do prohlížeče pomocí PHP funkce *flush()*, například informování uživatele o vytváření souboru. Protože i internetové prohlížeče mají omezenou dobu čekání na odpověď od serveru, a v případě, kdy prohlížeč nedostane po nějakou dobu žádná data, může být čekání na výsledek skriptu ukončeno. Po dokončení dokumentu je potřeba vyvolat přesměrování stránky prohlížeče, nebo vypsát na aktuální stránku odkaz, který vede k tomuto souboru.

3.2.2. Meziplatformní kompatibilita

Třída FPDF je napsaná v PHP, jedná se tedy o serverové řešení, a je nezávislá na platformě uživatele. Vyžaduje pouze, aby na serveru bylo nainstalováno PHP minimálně verze 4. FPDF verze 1.4 pracuje i s PHP3, ale jelikož se jedná o starší verzi této třídy, není doporučováno ji používat. Pro podporu GIF souborů je nutná knihovna GD, která umožňuje jazyku PHP pracovat s obrázky a animacemi typu GIF. Pro kompresi exportovaných souborů je nutná podpora ZLIB knihovny, určená právě pro kompresi souborů. Díky této knihovně mohou být výsledné soubory zmenšeny až na polovinu původní velikosti bez komprese. Testovaný soubor měl při vypnuté kompresi velikost 93,9 kB, při zapnuté vznikl soubor o velikosti 49,5 kB. Soubor obsahoval pouze vygenerovaný text Lorem Ipsum.

Problém výstupu při tvorbě PDF může nastat v internetovém prohlížeči Internet Explorer, který používá pro spojení s Adobe Readerem plugin obsahující mnoho chyb. V těchto prohlížečích je možné, že se exportovaný soubor vůbec nezobrazí. Tento

problém je možné vyřešit na straně serveru, a to tak, že exportovaný soubor bude uložen na serveru, a uživatel na něj přesměrován. V případě řešení na straně klienta je potřeba, aby uživatel zakázal zobrazování PDF souborů v prohlížeči Internet Explorer. Po tomto nastavení nebude exportovaný soubor zobrazen, ale odeslán na stranu klienta, kde bude i uložen.

3.2.3. Vhodnost pro konkrétní oblasti použití

Knihovna FPDF je určena převážně pro jednoduché prezentace textových prvků. Ve svém výchozím nastavení obsahuje pouze 3 základní fonty: *Helvetica*, *Calibri* a *Courier*. Všechny tyto fonty je možné použít s jejich zvýrazněním - tučně, kurzívou, nebo kombinací obou zvýraznění. Třída FPDF není bez dalšího rozšíření vhodná pro české znaky, jelikož ve výchozím nastavení neobsahuje ani jeden přiložený font české znaky s diakritikou. Pro české znaky je nutno dodat fonty typu TrueType1 nebo Type1, které je potřeba převést do podporovaného formátu.

Tyto fonty je možné vygenerovat spolu s příslušnými soubory určené pro FPDF třídu pomocí online aplikací nebo pomocí přiložené třídy MakeFont. Vygenerování souboru pomocí třídy je značně složité, proto je vhodnější použít online aplikace, která udělá sama veškeré nastavení. Při generování pomocí těchto metod jsou vygenerované soubory totožné.

Vygenerováním nového fontu vzniknou 4 soubory:

PHP soubor pro připojení fontu

AFM soubor s metrickými údaji písma Adobe

Z soubor s archivem Unixu

T1A soubor.

Pro použití nových fontů je nutné všechny vygenerované soubory nakopírovat do adresáře `./font/`, který se nachází ve složce FPDF knihovny. Poté je nutné nově vytvořený font připojit, k tomu slouží funkce `$fpdf->addFont(název fontu, styl, soubor)`, s parametry název fontu, který bude ve skriptu používán, styl fontu – zvýraznění, kurzíva, zvýrazněná kurzíva, nebo font bez vykreslení. Posledním parametrem je soubor, ve kterém je font. Jedná se o vygenerovaný PHP soubor. Knihovna nepodporuje UTF-8 kódování, je tedy nutné zapisované znaky převádět. Například pomocí PHP

funkce *iconv(kódování řetězce, výstupní kódování řetězce, řetězec)* na kódování, ve kterém je nový soubor vypisován. Ve výchozím nastavení se jedná o ISO-8859-1.

Další nevýhodou této třídy jsou komplikovanější algoritmy pro zápis do PDF souboru. Například pro vykreslení tabulky je nutno znát šířku sloupce a výšku řádku pro vykreslení jedné položky v tabulce. Při exportování rozsáhlých tabulek s dynamickou šířkou to může znamenat značný problém. Výhodou této třídy je, že byla jedna z prvních, která umožňovala jednodušší zápis do souboru PDF než předkompilovaná knihovna PDFLib. Proto pro tuto třídu existuje několik rozšíření, které její funkcionalitu mohou značně vylepšit. Ať už se jedná o rozšíření pro optimalizaci paměti nebo pro zjednodušení exportu některých prvků či jejich výsledné zobrazení. Export rozsáhlých tabulek je i s tímto rozšířením problémový, protože je nutné ještě před generováním tabulky znát rozměry jednotlivých sloupců. Bylo vytvořeno rozšíření pro UTF-8 kódování. Spíše než rozšíření jde o přeprogramování celé třídy pro podporu tohoto kódování. Vznikla tedy nová třída, která je prezentována jako rozšíření této třídy.

4. HTML2(F)PDF

HTML2(F)PDF je třída určená pro export souborů do formátu PDF. Podporuje export HTML značek a jejich stylování pomocí kaskádových stylů. Původní verze třídy HTML2(F)PDF vycházela z třídy FPDF, proto se v nadpisu uvádí písmeno F v závorce. Novější verze jsou založeny na třídách FPDF a TCPDF. Tato třída implementuje všechny rozšíření, které byly vytvořeny pro FPDF třídu. Mimo těchto rozšíření má všechny shodné parametry jako třída FPDF. Jde tedy pouze o rozšíření FPDF třídy, nikoliv o nový software. Třídu není potřeba nijak instalovat, stačí jí pouze nahrát na server do příslušného adresáře, a rozbalit.

4.1. Technická náročnost

Třída HTML2(F)PDF vychází převážně z FPDF třídy. Z hlediska technické náročnosti exportu textu je totožná jako původní FPDF třída. Jelikož se jedná o serverové řešení, neumožňuje export prvků, které jsou vygenerovány u klienta, například pomocí JavaScriptu, nebo jiných klientských skriptů, a export není závislý na platformě uživatele. Přidání nového fontu pro tuto třídu je shodné jako u FPDF třídy, tedy fonty typu TrueType1 nebo Type1. Po vygenerování souborů pro fonty je nutné připojit PHP soubor s novým fontem do skriptu, ve kterém je export použit. Dále stačí

pouze font připojit pomocí funkce třídy *AddFont()*. Třída na rozdíl od FPDF třídy podporuje UTF-8 kódování, a je nastaveno jako výchozí kódování pro tuto třídu. Pro správné zobrazení použitých znaků je potřeba mít obsah použitých znaků ve fontu. Výchozí fonty této třídy neobsahují české znaky.

4.2. Meziplatformní kompatibilita

Třída HTML2(F)PDF je založena na FPDF třídě, má tedy totožné požadavky a nároky. Jedná se o serverové řešení, není tedy závislé na platformě klienta, ani internetovém prohlížeči. Je vhodná pro stránky psané v jazyce HTML a CSS, ale i pro export textu s jednoduchým formátováním, stejně jako třída FPDF, ze které vychází. Z tohoto důvodu je potřeba GD knihovna pro podporu GIF obrázků, a knihovna ZLIB pro podporu komprese exportovaných souborů.

4.3. Vhodnost pro konkrétní oblasti použití

Třída HTML2(F)PDF je vhodná pro export jednoduchých webových aplikací. Jedná se o serverové řešení, které není závislé na platformě uživatele. Největší nevýhoda této třídy je, že při exportu často dochází k neidentifikovaným chybám, což z ní dělá značně nepoužitelný software. Výhodou je malá velikost této třídy a vysoká podpora HTML značek a vlastností kaskádových stylů.

Další nevýhodou je mnohočetný výskyt chybného zobrazení při exportu webové aplikace, které je způsobeno špatným vypočítáváním aktuální pozice daného prvku vzhledem k rozměrům exportované stránky. Může tedy dojít k přepisování již vyexportovaných řádků textu, nebo k neakceptování šířky stránky. Dojde tedy k vypsání některých prvků stránky mimo rozsah exportovaného souboru.

V příloze 1 jsou vidět chyby tohoto exportu, tedy přepisování řádků jednotlivých textových prvků exportované aplikace a nedodržení rozsahu stránky.

5. MPDF

MPDF je třída napsaná v jazyce PHP pod GPL licenci, určená pro přímý převod jazyka HTML do PDF. Umožňuje stylování exportovaného jazyka HTML pomocí kaskádových stylů s několika omezeními. Je implementováno i několik funkcí z CSS verze3, které současné prohlížeče prozatím podporují pouze částečně. Podporuje UTF-8

kódování a vychází ze tříd FPDF a HTML2(F)PDF s několika funkcemi navíc. Některé rozšíření pro FPDF třídu implementuje i MPDF, například průhlednost prvků v exportovaném souboru. Toto rozšíření vzniklo původně pro FPDF třídu, ale bylo upraveno pro lepší funkcionalitu a použito v rámci třídy MPDF. Pro formuláře je možnost akceptovat i JavaScript, který slouží ke kontrole a ovládání těchto formulářů. K podpoře JavaScriptu je nutné povolit aktivní exportované formuláře, ke kterým se JavaScript vztahuje a ovládá je.

Třída umožňuje 3 druhy zápisu jazyka webových aplikací pro export do PDF souboru:

- zápis samotného jazyka HTML, tedy pouze obsahu značky *body*.
- zápis celé stránky v HTML včetně hlaviček.
- zápis pouze kaskádových stylů pro formátování výstupu.

Při zápisu pouze kaskádových stylů je nutné zapsat i jazyk HTML, aby jednotlivé vlastnosti stylů bylo možné přiřadit k jednotlivým elementům stránky, a měly co stylovat. Pokud dojde k zápisu pouze kaskádových stylů bez HTML kódu, je vytvořen dokument s prázdnou stránkou.

5.1. Technická náročnost

PDF soubory jsou generovány v operační paměti RAM, což u velkých souborů může být značné omezení. Například při exportu rozsáhlých tabulek může dojít k překročení této paměti. Proto je dobré do PDF souboru zapisovat pouze data, která je potřeba zobrazovat v exportovaném souboru. Další možností je zapisovat data do tabulky bez kaskádových stylů, protože i formátování tabulek má značné paměťové nároky. Z doporučení MPDF je dobré vyhnout se obrázkům typu GIF, PNG s alfa kanálem, nebo prokládaným PNG. Vzhledem k náročnosti režie při exportu tabulek do PDF, a možnosti měnit v MPDF třídě parametry pro export tabulek, byla provedena měření při exportování tabulek s různým nastavením.

5.1.1. Měření exportu tabulek pomocí MPDF

Měření exportu tabulek bylo prováděno pomocí dat ze stránky <http://www.mapy-data.projekt-zdravi.cz>, která obsahuje záznamy o mapách z Libereckého kraje a má 131 záznamů. Pro větší rozdíly v měření byl kód stránky upraven, a obsah celé tabulky

vypsán desetkrát za sebou. Testovací stránka tedy obsahovala 1310 záznamů. Měření bylo zaměřeno na časovou a paměťovou náročnost exportu, a velikost výsledného souboru.

Ve výchozím nastavení MPDF knihovny trval export tabulky do PDF souboru 173,046 vteřiny, skript vyžadoval paměť 141MB RAM a výsledný soubor měl velikost 2,03MB.

Následovalo otestování při nastavené proměnné `$mpdf->packTable` na hodnotu *true*. V tomto nastavení jsou data z tabulek ukládána do souboru binárně. Při tomto nastavení skript využil paměť 93,25MB a trval 128,30 vteřin, velikost souboru se opět nezměnila, a zůstala na 2,03MB.

Další měření bylo provedeno s nastavením proměnné `$mpdf->simpleTables` na hodnotu *true*. Tato funkce tabulky „zjednoduší“, konkrétně zakáže komplexní ohraničení tabulky. Při nastavení `simpleTables` na *true* byla doba exportu 111,64 vteřiny, vyžadovaná paměť 81,5MB a vznikl soubor o velikosti 1,95MB.

Posledním zvoleným parametrem bylo nastavení proměnné třídy MPDF `$mpdf->cacheTables` na hodnotu *true*. Kdy dochází ke cachování tabulek v paměti a sníží se tím paměťové nároky exportu na úkor času. Při tomto nastavení trval export 200,46 vteřin, ale paměťové nároky se snížily téměř na jednu pětinu, konkrétně na 25,75MB RAM. Velikost výsledného souboru se nezměnila, a zůstala na původních 2,03MB.

Při nastavení všech výše zmíněných proměnných trval skript 208,11 vteřiny, potřeboval paměť 20,75MB a vytvořil soubor o velikosti 2,03MB. Z toho lze usuzovat, že nejvyšší prioritu má nastavení proměnné `$mpdf->cacheTables`.

Název	Abstrakt	Časový rozsah	Téma	Jazyk	Geografická ohraničení	Služba	Odpovědná organizace
Administrativní členění ČR	Administrativní hranice ČR, krajů, okresů, obcí s rozlišením pohraniční, katastrálních území	2007-05-25	hranice	CS	11.82950986213, 48.11208263333, 19.12764087445, 51.58831755561	prohlížení služba	ČÚZK
Adresní body a názvy ulic	Tato služba obsahuje vektoru adresních bodů a názvů ulic a veřejných prostranství ve formě detailních bodů.	2011-09-05	doprava	CS	11.614929, 48.144219, 19.277625, 51.487947	prohlížení služba	CENIA, česká informační agentura životního prostředí
ArčrČ 500	Tato mapová služba poskytuje digitální vektorové geografické databáze ArčrČ 500. Zahnuje sídla, silnice, lezebnice, vodní toky, vodní plochy a leze. ArčrČ 500 navazuje na podobné databáze, zpracované firmou ESRI nebo spolupracujícími firmami. Jejím cílem je zpřístupnění přehledných geografických informací o ČR uživateli geografických informačních systémů.	2011-09-05	obrazová data/základní mapy/ pokryv Země	CS	11.571252, 48.053485, 19.295833, 51.506889	prohlížení služba	CENIA, česká informační agentura životního prostředí
Automapa	Tato služba zobrazuje topografický podklad, který obsahuje kompletní síť silnic III. třídy, silnic II. třídy, lesní plochy, vodní toky a vodní plochy a lezebnice.	2011-09-05	doprava	CS	11.571252, 48.053485, 19.295833, 51.506889	prohlížení služba	CENIA, česká informační agentura životního prostředí
Celkový potenciál cestovního ruchu obcí Libereckého kraje v roce 2001	Mapa zobrazuje celkový potenciál cestovního ruchu obcí Libereckého kraje v roce 2001.	2001-01-01	společnost	CS		služba stabilizační dat	Krajská správa ČSÚ Liberec
CENIA přehledka	Tato mapová služba vyhledá čísla digitálních vektorové geografické databáze ArčrČ 500. Zahnuje vybraná sídla krajů, silnic, okresů a rozlišení pohraniční a okresní pohraniční hranice, základní síť silnic III. třídy, silnic II. a I. třídy. Během vyhledávání podléhá jako podklad silnic byla vymezena z výkresů DMU 23.	2011-09-05	obrazová data/základní mapy/ pokryv Země	CS	11.571252, 48.053485, 19.295833, 51.506889	prohlížení služba	CENIA, česká informační agentura životního prostředí
Chráněná území	Tato služba obsahuje vektor živelních území. 1) Registr chráněných lesních území, 2) Chráněná oblast přirozené akumulace vod (CHOPAV), 3) Plošnost správ CHKO 4) Biosferické rezervace, 5) Přírodní parky, 6) Územní systém ekologické stability.	2011-08-31	životní prostředí	CS	11.570753, 48.066135, 19.296195, 51.505703	prohlížení služba	CENIA, česká informační agentura životního prostředí
Chráněná oblast přirozené akumulace vod	Chráněná oblast přirozené akumulace vod (CHOPAV) jsou § 28 zákona č. 254/2001 Sb. o vodách a o změně některých zákonů (vodní zákon) definovány jako území, které pro své přírodní podmínky tvoří významnou přirozenou akumulaci vod. V těchto oblastech se zákonom č. 254/2001 Sb., v rozsahu stanoveném zákonem, vykonávají: (a) monitoring změn lesních porostů, (b) odvodnění lesní porosty, (c) odvodňování zemědělské půsokny, (d) sítě nádrží, (e) sítě zavlažování, (f) sítě ochrany povrchových zvláštností nebo povrchové územní úpravy, které by vedly k odmytí povrchové hladiny podzemních vod, (f) sítě a zpracování radioaktivní suroviny, (g) skladů radioaktivní odpady. Vlasta tyto oblasti vyhláškou zařazuje. Hranice těchto oblastí jsou vymezeny v náčrtových výkresy č.401/978 Sb., č.10/1979 Sb., č.85/1991 Sb. Evropské je vedené v rozsahu územní identifikace, popisu lesnic a názvů chráněné oblasti.	2005-12-15	životní prostředí	CS	12.1, 48.5, 18.9, 51.1	prohlížení služba	VUV TGM, v.v.i.
Cisáské povinné otisky stabilního katastru 1:2 880 - Čechy	Bázevné nastupé kopie tzv. cisáských povinných otisků map stabilního katastru Čech. Jedná se o mapy z let: 1826-1843, původně určené k archivaci v Centrálním archivu pozemkového katastru ve Vídni, odkud byly po vzniku Československé republiky v rámci archivu rozšířeny předáky do Prahy. Na rozdíl od tzv. originálních map stabilního katastru zachycují stav krajiny bez detailnějšího zápisu pozemských hranic. Dříve patřil ke straně badatelů se sejdáním a nepřevzatým archivním ÚAČK. Pro území Čech je archivováno cca 8400 katastrálních map na ca 31 tisících mapových listech. V katastrálních územích, pro která se tyto mapy nedochovaly, jsou postupně nahrazeny originálními mapami stabilního katastru.	2010-12-01	plánování/katastr	DE	12.09, 48.55, 18.86, 51.06	služba stabilizační dat	ČÚZK
CORINE Land Cover - zemědělských ploch mezi lety 1990 a 2000	Tato služba zobrazuje klasifikaci zemědělských ploch CORINE Land Cover 2000 a vektoru zemědělské databáze, která identifikuje plochy v rámci krajiny pokryvané v databázi CLC90 a CLC2000 (přehledy a detaily ploch).	2011-09-22	zemědělství	CS	11.571252, 48.053485, 19.295833, 51.506889	prohlížení služba	CENIA, česká informační agentura životního prostředí

Obrázek 3 - Exportovaná tabulka pomocí MPDF

Při měření není zmíněna HTML2(F)PDF třída, jelikož při exportu rozsáhlých tabulek dochází ke dříve zmíněné nedefinované chybě. Pro export pomocí této třídy byla dle doporučení připojena paměť 1GB RAM, nastavení času pro běh skriptu na 5 minut, a celý soubor po načtení konvertován do WINDOWS-1250 kódování pomocí PHP funkce *iconv()*. Ani toto nastavení nepomohlo předcházet nedefinované chybě. Jedná se tedy o jednu ze softwarových chyb třídy HTML2(F)PDF.

5.1.2. Porovnání velikosti exportovaného souboru z MPDF a FPDF

Jelikož MPDF třída používá kódování UTF-8, velikost exportovaných souborů je větší než u FPDF třídy, která má primárně nastavené kódování ISO-8859-1. Při testu byl použit soubor s obsahem 200 000 znaků bez diakritiky v kódování ANSI. Text byl vygenerován pomocí textového generátoru Lorem Ipsum. FPDF třída pro export potřebovala 1,75MB RAM a výsledný soubor měl velikost 154 kB (158 545 bajtů). Exportovaný soubor z MPDF knihovny měl 195 kB (199 853 bajtů), export vyžadoval paměť 3 MB. Pro zápis byly použity funkce *MultiCell(\$width, \$height, \$text)*, kterou mají obě knihovny totožnou, neboť MPDF vychází z FPDF. Ve třídě MPDF je tato funkce upravena, aby exportovaný řetězec konvertovala do UTF-8 kódování.

5.2. Meziplatformní kompatibilita

Jelikož se jedná o serverové řešení, není třída závislá na platformě uživatele. Pro její používání je vyžadováno pouze stažení z internetu a nahrání na server. Problém může nastat při zobrazování vygenerovaného PDF souboru v internetovém prohlížeči Internet Explorer, protože tento prohlížeč používá pro spojení s Adobe Readerem plugin, který obsahuje chyby.

Třidu je vhodné rozbalit do adresáře *./mpdf*. K instalaci stačí pouhé rozbalení komprimované třídy a nakopírování do příslušného adresáře na serveru. Po rozbalení je nutné zkontrolovat, zda jsou přiřazena oprávnění k zápisu pro složky *./Tfontdata*, *./Tmp* a *./Graph_cache*. Pokud chce uživatel otestovat funkce této třídy, je vhodné přejít v internetovém prohlížeči do složky *./mpdf/examples/* na doméně s MPDF třídou, kde jsou uvedeny příklady použití této třídy a jejich vygenerované PDF soubory.

Třída má ve svém výchozím nastavení velikost 1278KB (jedná se pouze o třídu, nikoliv o pomocné utility) a pro nahrání třídy do paměti je potřeba 7MB RAM. V případě, kdy dochází na straně serveru k několika exportům najednou, může tato velikost značně přesáhnout přidělenou kapacitu paměti. Nemusí pak dojít k dokončení žádného z aktuálně generovaných souborů. V tomto případě je možné využít verzi MPDF Lite, která snižuje paměťové nároky třídy při exportu. Pro tuto verzi je nutné nastavit některé vlastnosti ve třídě MPDF. Ke konfiguraci MPDF třídy je ve výchozí instalaci soubor *compress.php*, který je umístěn v kořenové složce MPDF třídy.

Pro editování funkcí třídy je nutné změnit název souboru *mpdf.php* na *mpdf_source.php*. Tento soubor slouží k upravování výsledné třídy MPDF. Poté je nutné v internetovém prohlížeči otevřít stránku se souborem *compress.php*. Po načtení stránky je uživateli umožněno změnit parametry a funkce MPDF třídy. Je umožněno pouze snížení paměťových nároků, nikoliv rozšíření funkcí MPDF třídy. Po stažení má třída maximální funkcionalitu. Podporuje tedy všechny funkce, které jsou v rámci této třídy implementovány.

Je možné změnit parametry třídy, například akceptování HTML+CSS, DIRECTW, zabarvení pozadí prvků, tabulky, obrázky, formuláře – ať už aktivní nebo ne, podporu fontů, atd. K volbě je 30 parametrů. Pro použití této třídy je potřeba podpora buď HTML+CSS nebo DIRECTW. HTML+CSS je potřebné pro zápis HTML

značek a kaskádových stylů. Tato podpora je nutná například pro zápis tabulek, seznamů a všech značek jazyka HTML a vlastností kaskádových stylů. Pro podporu exportů všech značek a vlastností pro HTML a CSS je potřeba ve třídě zaškrtnout podporu pro HTML+CSS, tabulky, seznamy, obrázky všech podporovaných formátů této třídy, formuláře, barvu pozadí, importy (pro stránky s externími kaskádovými styly), vykreslování rámečků prvků a podporu fontů. Při tomto nastavení byla výsledná velikost třídy 1,110KB a vyžadovala paměť 6,25MB RAM při nahrání do paměti.

Akceptováním pouze DIRECTW se MPDF třída vrátí do nastavení FPDF třídy, se všemi jejími funkcemi. Neakceptuje HTML značky, a má pouze základní funkce pro zapisování do PDF souboru. Jde tedy o zapisování pouze textu a jednoduchých grafických prvků, ale s podporou UTF-8 kódování. Při tomto nastavení měl výsledný soubor velikost 322kB a vyžadoval 2,25MB.

Při použití DIRECTW a dalších funkcí třídy MPDF vznikají nové možnosti exportu, které není možné v jazyce HTML vytvořit, například záhlaví a zápatí jednotlivých stránek, sloupce, čárové kódy, generování obsahu exportovaného souboru, atd. Při této podpoře měla výsledná MPDF třída velikost 420KB a vyžadovala 2,75MB RAM.

Z tohoto testování a zkoumání třídy MPDF lze odvodit, že podpora jazyka HTML značně zvyšuje paměťové nároky. Na požadavcích se nejvíce podílejí tabulky, které ve výsledném souboru zabírají 292KB, a vyžadují paměť 1MB RAM. Žádná jiná volitelná možnost této třídy takovéto požadavky nemá.

Po potvrzení změn je vytvořen nový soubor s třídou MPDF – *mpdf.php*, který obsahuje pouze požadované funkce této třídy. Tyto změny jsou nevratné, pro opětovné používání funkcí této třídy je potřeba buď znovu stáhnout tuto třídu, nebo zachovat soubor *mpdf_source.php*. Soubor *mpdf_source.php* by měl vycházet z původní třídy MPDF se všemi implementovanými funkcemi, které tato třída obsahuje ve svém výchozím nastavení. Paměťové nároky byly měřeny vytvořením nové instance této třídy. Tato režie se značně změní při zápisu dat určeným k exportu do PDF souboru pomocí MPDF třídy.

Příloha 9 obsahuje tabulku, která zobrazuje volby funkcionality všech volitelných parametrů třídy MPDF, dále jejich velikost na serveru, vytíženost serveru při použití a popis, co daný parametr znamená.

5.3. Rozdílné zobrazení mezi exportem pomocí MPDF a v prohlížečích

PDF soubory neumí zobrazovat jazyk HTML. V případě pokusu o zápis dojde k zapsání celého kódu, který je odeslán klientovi. Včetně HTML značek v souboru se všemi atributy, hlavičkami, JavaScriptovými skripty zapsanými v souboru, kaskádovými styly, atd. Proto je nutné tyto značky konvertovat a nastavit jim příslušné vlastnosti pro zápis do PDF souboru, například velikost písma u nadpisů nebo nastavení rozměru obrázků. Z tohoto důvodu není možné dosáhnout naprosto stejného zobrazení exportované stránky v exportovaném souboru jako v internetovém prohlížeči.

5.3.1. Formuláře pomocí MPDF

Rozdíl zobrazení mezi internetovými prohlížeči a exportovanými daty do PDF souboru je vidět především u jednotlivých položek formulářů. I když grafické zobrazení formulářů je závislé na prohlížeči, PDF soubory mají vlastní definované formuláře, které se z grafického hlediska neshodují s žádným internetovým prohlížečem, ale jejich funkcionality je shodná. MPDF třída podporuje všechny druhy editovatelných inputů z HTML 4. Výjimku tvoří tlačítko, definované značkou *button*, které v jazyce HTML slouží k připojení JavaScriptu, nejčastěji reakce na událost *onClick*. Při exportu této značky dojde pouze k vypsání textu, který je zobrazen na tlačítku. Toto tlačítko je možné nahradit pomocí zápisu: `<input type="button">`. Inputy z HTML 5, například `input` pouze pro emailovou adresu nebo datum, nejsou podporovány, protože tyto druhy inputů nejsou definovány ve specifikaci PDF. Jejich podpora v prohlížečích je prozatím nedostatečná.

MPDF akceptuje 2 druhy formulářů. Aktivní formuláře, určené k editaci dat. A neaktivní formuláře, ve kterých zapsaná data nelze změnit. Pro aktivní formuláře je potřeba nastavit veřejnou proměnnou `$mpdf->useActiveForms` na hodnotu *true*. Tato proměnná je primárně nastavena na hodnotu *false*.

Neaktivní formuláře jsou určeny především pro tiskový výstup. V exportovaném souboru jsou dobře viditelné, ale hodnota v nich zapsaná nemůže být po exportu do PDF souboru editována. Jde tedy o vykreslení obdélníku kolem textu, jehož hodnota je

zapsána v daném inputu, popřípadě vykreslení kolečka u inputu typu *radio*, který slouží k vybrání pouze jedné volby z několika možností, nebo vykreslení čtverce u inputu typu *checkbox*, který slouží k zaškrtnutí několika možností. U neaktivních formulářů není nutné vyplňovat atribut *name* pro jednotlivá políčka formuláře, který slouží k identifikaci dat ve formuláři při jejich následném odeslání nebo zpracování. Toto nastavení je tedy vhodné především pro tisk vyexportovaného dokumentu, nikoliv pro vytvoření souboru, který bude dále používán pro vyplňování formulářů na internetu.

Aktivní formuláře jsou určeny pro export webové aplikace a uchování souboru v elektronické podobě. Ve vygenerovaném PDF souboru je možné data v těchto formulářích měnit. U přepínače (atribut *type="radio"*) jsou automaticky měněny hodnoty, stejně, jako v internetovém prohlížeči. Takto je možné zaškrtnávat i zatrhávací políčko (*type="checkbox"*). Akceptováno je tlačítko *reset*, které po reakci na kliknutí vrátí hodnoty ve formuláři do původního nastavení. Další podporované tlačítko je typu *submit*, které po stisknutí odešle data z formuláře, popřípadě zavolá klientský skript pro kontrolu vyplněných dat ve formuláři. Jedná se o stejný princip jako odesílání formulářů pomocí HTML – data jsou odeslána na stránku, která je uvedena ve značce ohraničující celý formulář – *form*, v hodnotě atributu *action*.

Při použití aktivních formulářů je podmínkou, aby v HTML kódu exportované aplikace měl každý input ve formuláři definovaný atribut *name*, který slouží pro identifikaci odeslaných dat z formuláře. V případě, že tento atribut není definován, export je ukončen chybovou hláškou, kde je uživatel upozorněn na chybějící atribut. Po uložení vyexportovaného souboru nejsou změněné hodnoty ve formuláři uloženy, a formulář zůstane ve výchozím nastavení. Jsou v něm uvedeny tytéž hodnoty, které byly zapsány při exportu webové aplikace do PDF souboru. Po otevření tohoto souboru, například v programu Adobe Reader, a pokusu o změnu hodnoty ve formuláři je uživatel upozorněn na tento fakt, a je mu doporučeno následné vytisknutí celého souboru.

V příloze 6 je zobrazen rozdíl formulářů v internetovém prohlížeči a vyexportovaném souboru

5.3.2. Jednotky a rozdíl velikostí

Internetové prohlížeče akceptují několik druhů jednotek, zapsaných v HTML kódu stránky, popřípadě pomocí kaskádových stylů. Umožňují zápis rozměrů v absolutních i relativních jednotkách. Absolutní jednotky mají pevně danou velikost, která je běžně používána. Jedná se tedy o pixely (px) milimetry (mm), centimetry (cm), palce (in), typografický bod (tp) a pica (pc). Tyto jednotky mají pevně daný poměr mezi sebou a jsou vždy konkrétního rozměru, zatímco relativní jednotky jsou velikost písma (em), výška malého písmene x (ex) a velikost základního písma (rem). Velikost rem je dána velikostí písma v základním nastavení stránky, zatímco em je dána velikostí svého předka. Další používané jednotky jsou procenta, které přepočítávají velikost buď z původního nastavení elementu, rozměrů rodičovského elementu, nebo z šířky a výšky okna.

MPDF třída používá při exportu pouze jednotky v milimetrech. Všechny jednotky jsou tedy přepočítány, a před exportem převedeny na milimetry, a zaokrouhleny na celá čísla. Podporovány jsou všechny rozměry, které je možné zadat v jazyce HTML nebo CSS. Pokud je uveden rozměr bez jednotky, třída s tímto rozměrem pracuje stejně jako s hodnotou zadanou v pixelech.

Z tohoto důvodu může dojít k odlišnému vykreslení v internetovém prohlížeči a exportovaných dat v PDF souboru. Například nadpis H1 má ve svém výchozím nastavení prohlížeče velikost 24px. Při exportu třída MPDF tento rozměr převede na milimetry, tedy 6,349mm. A poté je převeden na celé jednotky, tedy na 7mm. V případě exportu rozsáhlých textových stránek může dojít k rozdílnému zobrazení v délce exportu, například o celou stránku A4.

V příloze 5 je vidět téměř shodné zobrazení s internetovým prohlížečem Google Chrome s několika výjimkami. Především se jedná o změnu rozsahu exportovaného souboru, který je způsoben přepočtem jednotek.

5.4. HTML + CSS

MPDF třída podporuje všechny značky HTML 4 a jejich zobrazení v exportovaném PDF souboru je téměř shodné jako zobrazení v internetovém prohlížeči. Podporovány jsou i některé značky z HTML 5, jejichž podpora je v prohlížečích prozatím nedostatečná. Podpora kaskádových stylů verze 2 je plně

akceptována, až na několik výjimek, například nejsou podporovány všechny druhy rámečků – vlastnost *border*, konkrétně hodnota *double*, která způsobí zdvojení rámečku kolem daného elementu na stránce. Vlastnost s touto hodnotou je zobrazena shodně jako hodnota *solid*, tedy jednoduchý rámeček zobrazený jako plná nepřerušovaná čára definované šířky.

5.4.1. Druhy zápisu HTML/CSS

Pro zapsání jazyka HTML se používá funkce *writeHTML(\$string, \$mode)*, u které parametr *\$string* určuje řetězec, který má být exportován do PDF. Parametr *\$mode* určuje druh zápisu proměnné *\$string*.

Pokud má parametr *\$mode* hodnotu 0, jde o zapsání textu včetně hlaviček souboru. Tento druh zápisu lze použít při exportu celých stránek. V tomto případě MPDF třída použije i kaskádové styly importované pomocí značky *<link* s atributem *media* definovaný pro hodnoty *all* nebo *print*. Ale MPDF třída ne vždy tyto styly načte. Pokud je totiž cesta zapsána relativně, například *../directory/file.css*, třída tuto cestu k souboru bere vzhledem ke svému uložení na serveru, nikoliv k pozici načítané stránky. Pokud je stránka načítána z jiného serveru, soubory nebudou načteny. Další kaskádový styl, s kterým je v tomto módu pracováno, je inline styl, který je definován značkou *<style></style>*. Poslední typ kaskádového stylu, s kterým tato třída pracuje v tomto nastavení, je atribut *style* u všech značek v načteném souboru. Tento zápis je vhodné použít v případě exportu celé stránky, včetně stylů určených pro tiskový výstup z aplikace.

Při hodnotě 1 parametru *\$mode* je pracováno s řetězcem, jako se zvlášť zapsaným kaskádovým stylem. V tomto případě je nutné do MPDF třídy zapsat i HTML, ke kterému má být styl připojen. Pokud je zapsán i HTML s hlavičkami, nedojde k načtení souborů připojených pomocí značky *<link>*. Tento druh zápisu je vhodný použít, pokud byl vytvořen odlišný styl pro export do formátu PDF, než pro tiskový výstup této aplikace. Zápis kaskádového stylu do souboru musí být zapsán před zápisem jazyka HTML.

Pokud má parametr hodnotu 2, jde pouze o zápis HTML kódu bez hlavičky souboru. Např. do proměnné *\$string* je tedy možné zapsat pouze „*<p>text</p>*“, není potřeba použít značku *body* na začátku a na konci zapisovaného řetězce. V tomto módu

je pracováno pouze se styly, které jsou součástí zapsaných značek, tedy atributy *style*, popřípadě pokud je zapsán kaskádový styl pomocí *writeHTML(\$styl, 1)*, který se k tomuto HTML kódu připojí. Toto nastavení je vhodné pro zápis rozšíření pro MPDF třídu pomocí značek, například definování záhlaví a zápatí stránek. Záhlaví a zápatí pro MPDF třídu je možné definovat pomocí značek určených pouze pro třídu MPDF.

Hodnota *\$mode* 4 a 5 slouží pouze pro interní použití. Při hodnotě 4 je kód pouze analyzován, ale není určen k žádnému výstupu. Při hodnotě 5 je ukládán do vyrovnávací paměti.

```
$url="http://www.nti.tul.cz";  
$string=file_get_contents(url);  
//vytvoření nové instance třídy MPDF  
$mpdf=new mpdf();  
//nastavení výchozí adresy, vzhledem k exportu externí stránky  
$mpdf->setBasePath(url);  
//zápis celého HTML kódu stránky  
$mpdf->writeHTML($string,1);  
//soubor bude po vytvoření zobrazen v prohlížeči  
$mpdf->output();
```

Třída při přidělování vlastností kaskádových stylů pracuje shodně, jako Internetové prohlížeče. Pokud tedy není vlastnost nikde definována, bere se výchozí hodnota. Pokud je vlastnost definována ve stylech vícekrát, je zobrazena poslední z nich. Vlastnosti rodiče se dědí na potomka a dle priority vnitřního elementu, tedy atribut *style* u značky přepíše vlastnosti v interním stylu i v načteném externím souboru s kaskádovým stylem. Třída MPDF nepodporuje klazuli *!Important*, která přidává vlastnosti vyšší priority, a není později přepsána.

Příloha 6 ukazuje použití několika vlastností kaskádových stylů, včetně vlastností kaskádových stylů verze 3.

5.4.2. Podpora HTML

MPDF podporuje téměř všechny značky HTML 4 a několik značek z HTML 5. Pokud je exportována značka, kterou třída MPDF nepodporuje, jsou ignorovány

všechny její atributy. Jediný použitý atribut je *style*, který je použit pro upravení vlastností obsahu nepodporované značky. Výjimku tvoří značka *dir* a *menu*, které jsou sice téměř nepoužívány a v HTML 5 již nejsou zmíněny, ale při exportu vzniká chyba, která je zobrazena na Obrázku 4. Nejenže obsah této značky není při exportu zobrazen, ale dojde i k absolutnímu skrytí obsahu několika následujících značek, a další značka, která je na stránce zobrazena, má chybné formátování.

HTML

- Položka v seznamu tagu *dir*

MPDF

První termín tag *dt* Výklad prvního termínu tag *dd*
Druhý termín tag *dt* Výklad druhého termínu tag *dd*

Nadpis H2

- Položka v seznamu tag *ul*

První termín tag *dt*

Výklad prvního termínu tag *dd*

Druhý termín tag *dt*

Výklad druhého termínu tag *dd*

Obrázek 4 - Chybné zobrazení seznamu značkou *dir*

Třída MPDF nepodporuje rámy, které nejsou pro tiskový výstup vhodné.

Obrázky a odkazy by měly mít při exportu uvedenou absolutní cestu, protože třída používá relativní cestu vzhledem ke svému umístění, nikoliv k umístění exportované stránky. V případě, že je MPDF třída ve stejném adresáři jako exportovaný soubor, není potřeba toto omezení dodržet. Pokud je ale exportovaný soubor stažen ke klientovi, u odkazů je nutné uvést absolutní cestu. V opačném případě by byl uživatel po kliknutí na odkaz přesměrován na stránku s relativní cestou, která pravděpodobně nebude existovat. Tomuto omezení se dá předcházet i změnou hodnoty *\$mpdf->SetBasePath(URL);*. Tímto příkazem je nastavena základní URL pro všechny relativní cesty v exportovaném souboru, včetně externích kaskádových stylů, odkazů, cest k obrázkům, atd. Dojde však ke zpomalení celého exportu.

Pro MPDF třídu je možné definovat speciální HTML komentář zápisem *<!-- mpdf „kód pro MPDF“ mpdf-->*. Jde o podobné zapsání, jako komentáře pro Internet Explorer. Obsah tohoto komentáře akceptuje pouze třída MPDF. Všechny prohlížeče ho ignorují.

V přílohách 2, 3 a 4 jsou zobrazeny exporty vnořených tabulek, textu s logickým a fyzickým formátováním a různé možnosti rámečku tabulek.

V příloze 5 je zobrazen reálný export stránky v HTML pomocí MPDF třídy a porovnán s tiskovým výstupem z prohlížeče.

5.5. Fonty, cizí znaky a zrychlení exportu

MPDF třída obsahuje ve svém výchozím nastavení 46 fontů. Jedná se o několik standardních písmen, jako jsou Arial/Helvetica, Times a Courier, ZapfDingbats a „Symbol character set“ pro speciální znaky. Tyto fonty by měly být ve všech výchozích instalacích softwarů pro prohlížení PDF souborů, a nemusí být připojeny k exportovanému PDF souboru pro správné zobrazení znaků. V tomto případě je doporučována konfigurace MPDF knihovny pomocí parametru v konstruktoru *c*. Při tomto nastavení dojde k rychlejšímu exportu stránky, minimální velikosti exportovaného souboru a minimálnímu využití paměti. Výsledný soubor by se měl zobrazovat ve všech PDF prohlížečích shodně. Nevýhodou je omezení výběru písma, jelikož je omezeno pouze na standardní písma.

V případě, že jsou použity fonty, u kterých není zaručeno správné zobrazení u uživatele, například fonty CJK (chinese-japanese-korean), je vhodné odkazovat na stáhnutí těchto fontů. Tyto fonty je možné stáhnout přímo na stránkách společnosti Adobe v souboru „fonts-pack“. V tomto souboru jsou obsaženy 4 asijské fonty, které obsahují všechny kódové znaky. Jedná se o fonty, které jsou obsaženy ve výchozí instalaci MPDF třídy. V případě rozšíření znakové sady MPDF knihovny je doporučeno uživatele odkazovat na stránky, kde lze použité fonty stáhnout.

Poslední možností prací s fonty je přidat znakovou sadu do souboru. V tomto případě je nutné použít parametr v konstruktoru *S*. Velikost exportovaného souboru v tomto nastavení vzroste, zvýší se nároky na paměť a doba exportu se prodlouží. V případě, že je použito méně než 30% znaků dané sady, jsou přidány pouze použité znaky při exportu. Pokud je použito více znaků, je přidán celý font. 30% je výchozí nastavení MPDF třídy a lze ho změnit pomocí proměnné *\$this->percentSubset*, do které je zadáno procentuální zastoupení znaků v exportovaném souboru. V tomto případě dojde sice ke zmenšení velikosti exportovaného souboru, ale zároveň naroste doba zpracování, jelikož třída musí vybrat jednotlivé znaky, které budou postupně přidávány do použité podmnožiny.

Přidání nových fontů do MPDF třídy je rozdílné od FPDF a HTML2(F)PDF třídy. U MPDF třídy není možné nový font přidat pomocí funkce do skriptu při generování PDF souboru. Soubor s fontem musí být TrueType font typu TTF, který je potřeba nahrát do adresáře *./tfonts*. Akceptovány jsou i TrueType Collection – TCP a OpenType files – OTF v TrueType formátu. Dále je nutno připsat kód v konfiguračním souboru pro fonty – *config_fonts.php*, kde jsou všechny použité znakové sady definovány v poli.

```
$this->fontdata=array(
...
//definování názvu nového fontu
("ARIAL" => array(
//cesta k fontu
'R' => "ARIAL.ttf",
'sip-ext' => 'sun-extb',
),
... );
```

Po přidání tohoto kódu a jeho uložení je možné při exportu použít font pomocí kaskádového stylu, například atributem u textové značky *style="font-family:ARIAL"*.

5.6. Export blokových elementů

MPDF třída plně nepodporuje vnořené blokové elementy. Například blokový element, značka *div*, nemůže být v jiné značce *div*. V případě tohoto vnoření je šířka vnitřní značky přepsána na rozměr vnější.



Obrázek 6 - Dva vnořené blokové elementy

Na obrázku 6 je vidět, jak jsou implementovány 2 vnořené značky typu *div* s definovanou šířkou – *width*, výškou - *height*. Vnitřní značka *div* má navíc definovány vlastnosti *margin*, *padding*, *border* a *background*, zatímco vnější pouze vlastnost *background*. Při tomto zobrazení byl použit interní kaskádový styl, který byl definován

v hlavičce exportovaného souboru. Rozdílnému zobrazení vnořených blokových elementů lze předejít vnořeným stylem u blokového elementu, pomocí atributu *style* u vnitřního blokového elementu, kde budou zapsány vlastnosti tohoto bloku. V takovém případě dojde ke shodnému zobrazení, jako v internetovém prohlížeči.

Ukázka exportu reálné stránky je zobrazen v příloze 5.

5.7. JavaScript

Třída MPDF podporuje JavaScript pro aktivní formuláře. JavaScript je možné exportovat přímo z HTML nebo pomocí speciální funkce *SetJS(\$javascript)*. Pro podporu JavaScriptu u aktivních formulářů je nutno nastavit MPDF proměnnou *\$mpdf->useActiveForms* na hodnotu *true*. Po tomto nastavení se formuláře změní z obrázkových prvků v textu na plně ovladatelné textové pole. Pro ovládání aktivních formulářů pomocí JavaScriptu je nutné použít „Acrobat“ JavaScript, který se mírně liší od JavaScriptu, který implementují prohlížeče.

JavaScript je možné použít pro kontrolu dat ve formuláři. V tomto případě je nutné, aby každé editovatelné pole ve formuláři mělo přiřazený atribut *name*, který slouží k identifikaci jednotlivých položek formuláře, ať už pro ovládání JavaScriptu nebo pro odeslání dat na server. Pro odeslání dat serveru je potřeba, aby exportovaný HTML kód obsahoval ve formuláři atribut *action*. V tomto atributu je uvedena URL adresa, na kterou mají být data z vyplněného formuláře odeslána. Zároveň je nutné uvést atribut *method*, který slouží k identifikaci, zda jsou data odesílána skrytě pomocí metody POST nebo v URL adrese metodou GET. Odeslaná data mohou být následně zpracována na serveru.

5.8. Barvy, obrázky a vektorová grafika

Knihovna používá všechny druhy zápisu barev, které využívá HTML a CSS, tedy zápis názvu barvy, např. *black*, hexadecimální zápis např. *#FFFFFF*, nebo *#FFF*, zápis RGB – *rgb(255,255,255)*. Jejich zobrazení je shodné jako v internetovém prohlížeči.

Je akceptováno několik různých formátů obrázků. Jedná se o obrázky formátu GIF, PNG, JPG, WMF, SVG, BMP a generované obrázky z PHP skriptů. Průhledné, prokládané a transparentní obrázky formátu GIF jsou také podporovány. Obrázky je

možné vykreslit přímo v obsahu stránky jako vodoznak nebo vykreslit na pozadí. Velikost obrázku je omezena na aktuální pozici na stránce nebo šířku nadřazeného elementu, ve kterém je obrázek vykreslen. Nejvhodnější formát obrázku pro export z hlediska rychlosti a velikosti exportovaného souboru je formát JPG a PNG bez alfa kanálu nebo neprokládaný PNG. Export obrázků formátu GIF způsobí značné zpomalení aplikace. Pokud se jedná o GIF animaci, je v exportu zobrazen první obrázek této animace. Pro export není vyžadována GD knihovna, vytvořená pro práci s těmito obrázky. Ale bez této knihovny je export GIF obrázků značně pomalejší. Při jejím použití dojde k výraznému nárůstu paměti, až na desetinásobek velikosti obrázku. Při exportu několika obrázků najednou tedy může dojít k nedostatku prostředků pro export webové aplikace do PDF souboru.

V případě uvedení relativní adresy k exportovanému obrázku, je brána pozice vzhledem k pozici třídy MPDF, nikoliv k exportované stránce. Pro tuto změnu je nutné nastavit URL adresu pro třídu MPDF pomocí `$mpdf->setBasePath('url')`. Po nastavení této proměnné je zohledňována zadaná URL adresa. Relativní cesty k obrázkům a kaskádovým stylům jsou používány dle zadané URL adresy, nikoliv dle pozice třídy MPDF na serveru.

Vektorová grafika je podporována v jazyce HTML od verze 5, a její podpora internetových prohlížečů v posledních verzích je již značná. MPDF třída tuto grafiku vykresluje téměř shodně jako internetové prohlížeče. Tuto grafiku nepřevádí do bitmapového formátu, jako jiné třídy pro export HTML do PDF. Navíc MPDF třída podporuje vlastnosti CSS 3 pro vykreslení přechodů barev, které nejsou v prohlížečích prozatím podporovány.

Příloha 7 zobrazuje export vektorové grafiky a značky HTML 5 – *meter* a *progress*.

5.9. Ovládací značky třídy MPDF

Třída implementuje i několik značek, které nejsou v HTML podporovány, a jsou určeny pouze pro úpravu stránek v PDF souboru. Tyto značky neumí prohlížeče zobrazit, protože nejsou v žádné specifikaci W3C. Akceptuje je pouze třída MPDF. Jejich zapsání do stránky způsobí nevalidní stránku. Tomu lze předejít komentováním značky do komentářů určených pouze pro MPDF třídu - `<!--mpdf zápis` pro MPDF

mpdf-->, nebo exportovat stránku v jazyce HTML, kde budou dodefinovány tyto značky. Tyto značky mohou být nahrazeny funkcemi přímo určenými pro zobrazení a formátování těchto značek ve vygenerovaném PDF souboru. Tyto funkce jsou vytvořeny v rámci MPDF třídy, která je implementuje. Použití značek je vhodné v rámci exportu jedné HTML stránky, zatímco použití funkcí při exportu je vhodné pro několik oddělených částí kódu.

5.9.1. Anotace

Třída umožňuje zapsat anotaci pro exportované prvky na stránce. Jsou 2 možnosti, jak ji zapsat, buď na stránku vložit značku *<annotation>*, ke které je možné přidat atributy jako text v anotaci, její předmět, pozici zobrazení, ikonu, autora anotace, barvu a způsob, kdy se anotace zobrazí. Druhou možností je zapsat anotaci přímo v PHP kódu pomocí funkce *Annotation()*, která používá stejné parametry jako značka atributy. Anotaci nelze zobrazit při náhledu v prohlížeči, pro její zkontrolování je potřeba nejdříve soubor stáhnout, a poté otevřít v některém z programů pro zobrazení PDF souborů, např. Adobe Reader.



Obrázek 7 - Anotace

5.9.2. EAN Čárové kódy

Zapsání a zobrazení čárových kódů je možné buď pomocí značky `<barcode>`, nebo funkcí. K zápisu pomocí PHP funkce je nutné



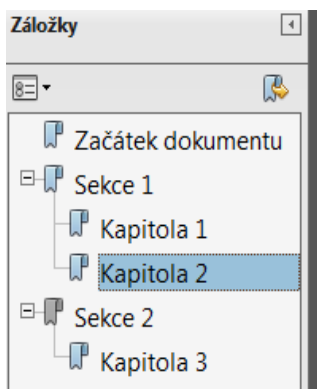
Obrázek 8 - Barcode

v kódu vytvořit novou instanci třídy `PDFBarcode()`, a zapsat do PDF funkcí této třídy

`getChecksum(číslo_kódu, typ)`. Značky `barcode` má povinný atribut `code`, do kterého je zapsáno číslo reprezentující čárový kód. Dalšími atributy jsou typ kódu, text, výška a šířka. Tyto atributy jsou nepovinné. Čárové kódy jsou zobrazeny jako svisle vykreslené čáry, nikoliv obrázek.

5.9.3. Záložky

Vytvořit záložky pomocí MPDF třídy lze opět pomocí atributů, nebo pomocí funkce. V PHP k tomu slouží funkce `bookmark()`, která má první parametr název záložky, a druhým parametrem je level této záložky. Tímto levelem je myšleno vnoření do záložek. Na obrázku mají Sekce level 0 a kapitoly level 1. Po zavolání funkce `bookmark()` je nutné zapsat do exportovaného souboru text, protože záložky se nevztahují k textu, ale k pozici dokumentu, kde byly v PHP vytvářeny. Text je možné zapsat například pomocí funkce `writeHTML(„Sekce 1 text“)`. Značka `bookmark` se vždy umísťuje před text, ke kterému má být záložka přidána a obsahuje stejné atributy, jako



Obrázek 9 - Záložky

Sekce 1 text
Kapitola 1 text
Kapitola 2 text
Sekce 2 text
Kapitola 3 text

parametry funkce `bookmark()`, tedy název záložky a její úroveň. Záložky je možné použít pouze po stažení souboru, nikoliv v náhledu dokumentu v prohlížeči. Jedná se o funkci prohlížečů PDF souborů. V případě použití těchto záložek a následného

stažení je soubor značně nestabilní. Po jeho otevření dochází k častému uzavření

prohlížeče Adobe Reader. Při exportu je tedy vhodné se těmito záložkami vyhnout.

5.9.4. Sloupce

Text ve výsledném PDF je možné rozdělit do sloupců. Nastavení sloupců je nutné udělat ještě před zápisem textu, který má být rozdělen, do třídy. Dělá se to funkcí

SetColumns(počet sloupců, zarovnání textu, vzdálenost mezi sloupci v mm), která způsobí rozdělení stránky na počet sloupců v zadaném parametru. Pokud je potřeba tyto sloupce zrušit, stačí zadat hodnotu 1. Pro přidání nového sloupce slouží funkce *AddColumn()*, která přidá pouze 1 sloupec. Pomocí značek je možné dosáhnout rozdělení stránky značkou *columns*, která má atributy pro počet sloupců zarovnání textu a vzdálenost mezi sloupci. Třída nemá funkci pro ukončení sloupce a začátek nového. Dá se ale definovat značkou *columnbreak*, která je bezparametrová a způsobí ukončení daného sloupce. V CSS 3 je možné definovat sloupce pomocí vlastnosti *column-count: počet-sloupců;*. Tato vlastnost prozatím není v prohlížečích téměř podporována, a ani MPDF třída ji nepodporuje. Sloupce v jazyce HTML lze také vyřešit pomocí zápisu 3 blokových elementů, které budou mít definovanou vlastnost *float*.

5.9.5. Tečkování

Pro seznam hodnot je možné využít tečkování. V MPDF se dělá značkou *dottab*, která je bez atributů, a způsobí tečkování od začátku řádku až na jeho konec, a automatické zalomení na konci řádku. Tuto značku není možné v HTML nijak nahradit.

Tag *dottab* bez atributů

Obrázek 10 – Tečkování značkou *dottab*

5.9.6. Ukončení stránky

Přidat novou stránku do vytvářeného dokumentu je možné pomocí značek *pagebreak* a *formfeed*. Rozdíl mezi nimi je ten, že *pagebreak* ve výchozím nastavení ukončí všechny otevřené značky, a vlastnosti pro stylování dokumentu uvede do původního nastavení, pokud nejsou definovány v attributech. Poté jsou automaticky přidávány stránky s tímto stylem až do konce dokumentu, nebo dokud nedojde k pozměnění hodnot.

Značka *formfeed* otevřené značky ani styl vykreslování na předchozí stránce nijak neukončuje, a pokračuje s jejich nastavením až do konce nebo jiného nastavení jejich hodnot.

Formfeed je tedy vhodné používat, pokud chceme na následující stránce změnit minimum vlastností, zatímco *pagebreak* je vhodné použít, pokud má mít nová stránka jiný styl než předchozí. Při použití této značky nedojde pouze k ukončení vlastností

kaskádových stylů, ale i ke zrušení záhlaví, zápatí a vynulování číslování stránek. Po použití této značky je nutné definovat vše, co je potřeba znovu použít.

5.9.7. Záhloví a zápatí stránky

Pro záhlaví a zápatí stránky používá MPDF třída téměř identické značky. Pro definici záhlaví a zápatí je možné využít HTML značky *pageheader* a *pagefooter*. Tyto značky mají atributy *content-left*, *content-center* a *content-right*. Slouží k zobrazení textů v záhlaví/zápatí na každé stránce. Každý atribut je určen pro konkrétní pozici. Pomocí speciálního zápisu, který je podporován pouze MPDF třídou, je možná do těchto atributů zapsat čísla stránek, datum generování, apod. Dalším atributem těchto značek je atribut *line*, který má hodnoty *on* a *off*. Jde tedy o zobrazení vodorovné čáry pod záhlavím/nad zápatím. Posledním atributem je *header-style* a *footer-style*, který umožňuje změny formátování záhlaví a zápatí, do kterého lze zapsat stejné hodnoty jako do atributu *style*. Jde tedy o zápis vlastností a hodnot jazyka CSS. U těchto značek je povinný atribut *name*, který slouží k pozdějšímu zavolání konkrétního záhlaví/zápatí.

Po definování značky *pageheader* a *pagefooter* je nutné do stránky zapsat značku, která zobrazí záhlaví a zápatí v exportu. K tomu slouží značky *setpageheader* a *setpagefooter*, které zobrazí záhlaví/zápatí v exportovaném souboru. Obě značky mají totožné atributy. Atribut *value* může nabývat hodnoty *on/off* a slouží k zobrazení / skrytí příslušného záhlaví a zápatí. Dalším atributem je *show-this-page* a hodnotou je číslo. Jde o číslo stránky, kde se záhlaví/zápatí zobrazí nebo skryje. V tomto případě závisí na hodnotě atributu *value*. Poslední atributem je atribut *name*, kterým se identifikuje příslušné záhlaví / zápatí. Poslední značkou pro definování záhlaví/zápatí jsou značky *sethtmlpageheader* a *sethtmlpagefooter*, které jsou totožné se značkami *setpageheader* nebo *setpagefooter*. Na rozdíl od nich však tyto značky podporují zápis jazyka HTML. Těmito značkami je možné do záhlaví/zápatí importovat obrázky, formátovat text, odkazy, seznamy, atd.

Příloha 6 zobrazuje použití záhlaví a zápatí na exportované stránce.

5.9.8. Vykreslení textu do kruhu

K tomuto vykreslení je potřeba použít značku *textcircle*, která má atributy *r* – poloměr kruhu, *top-text* – horní text, *bottom-text* – dolní text, *divider* – znak mezi horním a dolním textem, *space-width* – šířku mezery, *char-width* – šířku znaků v textu.

Ekvivalentem k této značce je funkce *CircularText()*, která vytvoří shodný výsledek. Na pozadí textu je vykreslena bloková značka *div* s definovanou vlastností *border-radius* pro zaoblení rohů. Značka *textcircle* není v prohlížečích nijak vykreslena, je tedy zobrazeno pouze pozadí prvku – značka *div*.

Obrázek 11 - Značka *textcircle* s vlastností *border-radius*



5.9.9. Obsah

Pro export rozsáhlých publikací je možné generovat obsah exportovaného souboru. K tomu je nutné definovat, co je potřeba do obsahu zahrnout, např. nadpisy, obrázky, přílohy. K definování položek zahrnutých v obsahu slouží značka *tocentry*, která má atributy *content* – sloužící k popisu položky obsahu, která je zobrazena ve vygenerovaném obsahu. Dalším atributem je atribut *level*, který nabývá celých číselných hodnot, a jde o úroveň seznamu. K vygenerování a zobrazení obsahu slouží značka *tocpagebreak*. Je bez atributů a v místě stránky, kde je umístěna, je při exportu vygenerován obsah.

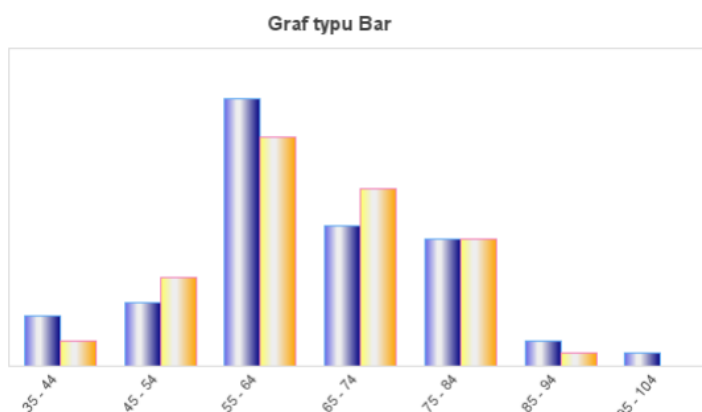
5.10. Rozšíření pro MPDF třídu a její vývoj

Pro MPDF třídu bylo vytvořeno několik rozšíření, které mohou její funkcionalitu značně vylepšit. Jelikož se jedná o projekt s velkým potenciálem v oblasti exportu webových aplikací do formátu PDF, stále dochází k inovaci této třídy a rozšíření jejích možností a funkcionality. I když v roce 2011 došlo na 6 měsíců k přerušení vývoje této třídy, tak kvůli jejímu velkému potenciálu a žádostem uživatelů, byl znovu obnoven.

5.10.1. Grafy

MPDF třída ve svém výchozím nastavení grafy nepodporuje. K jejich podpoře je vyžadováno rozšíření v podobě JpGraph knihovny, která je distribuována pod GPL licencí. Knihovnu lze stáhnout na stránkách <http://jpgraph.net/>. Z této knihovny je

potřeba rozbalit obsah adresáře `./src` do složky `./jgraph` v kořenovém adresáři MPDF třídy.



Obrázek 12 - Exportovaný graf pomocí značky `jgraph`

Pro použití samotných grafů je nutné nastavit proměnnou `$mpdf->userGraphs` na hodnotu `true`, která slouží k aktivaci podpory grafů při exportu. Dále je potřeba vytvořit tabulku, jejichž hodnoty budou zobrazeny v grafu. Je nutné,

aby tabulka s grafy byla na exportované stránce uvedena dříve, než značka určená pro zobrazení grafu. V opačném případě dojde k vypsání pouze tabulky s daty, a samotný graf zobrazen není. Tabulka je ke grafu připojena pomocí atributu `id`, který pro toto použití musí být definován.

Značka pro zobrazení grafů je `jgraph` a používá několik atributů. Prvním atributem je `table`, který určuje id tabulky, z které má být graf vytvořen. Následující atribut je `type`, který určuje, jaký typ grafu má být vykreslen. Je možné vykreslit sloupcový graf, sloupcový graf otočený o 90 stupňů, spojnicový, paprskový, výšečový, 3D výšečový, x-y bodový a „scatter“. Dalšími možnostmi nastavení dle atributů jsou rozlišení, titulek, vyhlazení linek pro x-y bodový graf, u sloupcového grafu zobrazování hodnot nad sebou nebo vedle sebe, zobrazení legendy ke grafu, její pozice a otočení, dále která data se mají v grafu zobrazit – od kterého řádku a sloupce, šířka a výška grafu, atd. Jediné, co tato značka postrádá, je možnost definování barev pro jednotlivé části grafu, protože jsou vykreslovány ve světlých barvách.

5.10.2. QR kódy

QR kódy slouží pro automatizovaný sběr dat. Funkce pro generování QR kódů z textu prozatím není implementována ve výchozím nastavení třídy MPDF. Jedná se pouze o samostatnou aplikaci s uživatelským rozhraním, která je ve fázi testování. Po zadání řetězce a potvrzení pro vygenerování příslušného QR kódu je vytvořena tabulka daného rozměru, která zobrazuje právě tento textový řetězec. Konkrétně se jedná o zobrazení tabulky, ve které má každý prvek rozměry 5*5px a barvu pozadí černou nebo

bílou dle zakódovaného řetězce. Vývojáři prozatím nejsou plně rozhodnutí, zda implementují tento algoritmus pro použití QR kódů, protože export tabulek může být z časového hlediska náročný. Uvažuje se i o generování obrázkových kódů v rámci této třídy, ale možnost generování pomocí tabulek bude jistě úspornější.

5.10.3. Vývoj třídy

Již první verze MPDF 1.1 podporovala UTF-8 kódování a byla plně otestována na PHP verzi 5. Její funkce bylo možné využít již od PHP verze 4.3. MPDF verze 5.4 a vyšší vyžaduje minimálně PHP verzi 4.3.1 nebo 5.0.3 z důvodů použití funkcí pro UTF-8 kódování.

Od MPDF 3.0 byly přidány vlastnosti pro barvu nebo obrázek na pozadí všech prvků, včetně vlastnosti CSS 3 *linear-gradient*, která vykreslí vodorovný přechod barev, nebo *radial-gradient* pro vykreslení kruhové přechodu. Od této verze je možné zapsat barvy všemi druhy zápisu, tedy názvem barvy, hexadecimálním zápisem nebo formátem RGB. Dále implementace vlastnosti *border-radius* pro zakulacení rohů rámečku prvků, nebo podpora PNG obrázků s alfa kanálem pouze pro obrázky v HTML, nikoliv pro formát těchto obrázků v pozadí. Další rozšíření se netýká jazyka HTML, ale výstupu PDF souboru, konkrétně definování záhlaví a zápatí stránek (pouze 1 pro celý dokument), a možnost přidání číslování stránek, včetně jeho nulování uprostřed exportovaného dokumentu.

Verze 4.0 třídy MPDF nepřinesla žádné významnější rozšíření ze strany podpory HTML a CSS. Došlo spíše k rozšíření podpory fontů, konkrétně pro indické a arabské znaky a přidání třídy na vytváření vlastních fontů, konkrétně jejich generování. Z jazyka CSS byla implementována hodnota *fixed* u vlastnosti *position*, která umístí prvek na přesně danou pozici, a díky této vlastnosti je možné navzájem překrývat jednotlivé blokové elementy, což v předchozích verzích nebylo možné. Zvýšila se podpora pro čárové kódy. Konkrétně je možné definovat více druhů těchto kódů.

Velké rozšíření pro podporu vlastností HTML a CSS přinesla až verze MPDF 4.3. Došlo k možnosti definování několika druhů pozadí na jedné stránce. Je možné tedy využít na pozadí jedné stránky obrázek včetně barvy pozadí, například přechod. Pro ohraničení blokového elementu vznikla podpora vlastnosti *border*: hodnoty *double*, která způsobí zdvojení rámečku kolem prvku. Při exportování textu s použitím CJK

fontů bylo nutné po stažení exportovaného PDF souboru uživatelem, aby si uživatel stáhl i potřebné fonty pro správné zobrazení těchto znaků. Od verze 4.3 je možné přidávat do exportovaného souboru pouze použité znaky. V předchozích verzích byla možnost přidat vždy pouze celé fonty, čímž se prodloužila doba exportu a velikost výsledného souboru. Od této verze je již možné definovat i obrázky PNG s alfa kanálem na pozadí. Pro obrázky v jazyce HTML přibyla vlastnost *padding* pro vnitřní odsazení obrázku od rámečku. Podpora pro obrázky generované z PHP skriptu a dědičnost z CSS pro velikosti. Od této verze je tedy možné zadávat všechny velikosti pomocí relativních velikostí v jazyce HTML a CSS, konkrétně jsou podporovány jednotky procenta, em, atd. které přepočítávají výslednou velikost prvku dle svého předka.

Velké rozšíření z hlediska použití fontů přinesla verze 5.0, a to možnost přidávat nové fonty typu TrueType1, a jejich čtení a vkládání přímo z TTF souboru. Do této verze bylo nutné přikládat ke třídě všechny vygenerované soubory, stejně jako u MPDF třídy. Od této verze stačí přidat pouze soubor TTF, a definovat ho v souboru *config_fonts.php*.

Ve verzi 5.1 došlo k vývoji především grafické podpory definování pozadí prvků a rámečků. Je možné exportovat celou stránku pouze ve stupních šedi pomocí definování proměnné *\$mpdf->restrictColorSpace=1*. Obrázky na pozadí u tabulek je možné definovat v rámci jednotlivých prvků tabulky, tedy pro řádky, nebo jednotlivé prvky stránky. Vykreslení rámečků je možné definovat pouze pro jednotlivé prvky tabulky, například řádky, a dále implementace vlastnosti kaskádových stylů verze 3 pro mozilla –*moz-linear-gradient*, která vykreslí gradient pouze v prohlížeči Mozilla Firefox. Vývoj přinesl i obecné rozšíření vlastnosti *linear-gradient* z hlediska vykreslení přechodu u obrázku, a možnost rotace tohoto obrázku. Byla rozšířena podpora pro číslování seznamů, konkrétně zobrazení čísel v jiném písmu, než v dekadickém, například římské číslice nebo arabské, popřípadě další Asijské písmo.

Od verze 5.3 byla vytvořena podpora pro aktivní formuláře a JavaScript. Pro funkci skriptu je potřeba ji psát JavaScriptem od firmy Adobe, a každé políčko formuláře musí mít definovaný atribut *name* pro konkrétní zaměření tohoto prvku pomocí JavaScriptu.

Ve verzi 5.4 došlo k rozšíření o možnost exportu obrázků vektorové grafiky, konkrétně vykreslení vektorů pomocí definice obecné značky *svg*, která je v HTML

podporována. V kaskádových stylech přibyla podpora pseudotřídy *:nth-child()*, díky které je možné definovat vlastnosti například pro každý sudý prvek. Z vlastností CSS 3 vyšla podpora pro vykreslení stínu pod všemi prvky, ať už pod písmem nebo pod celým blokovým elementem. Poslední vlastností rozšíření je vykreslení textu do kruhu pomocí značky *textcircle*.

Poslední verze třídy je 5.6, které přinesla rozšíření zejména z hlediska podpory značek HTML 5. Konkrétně jde o značky *article*, *aside*, *details*, *figure*, *figcaption*, *footer*, *header*, *hgroup*, *nav*, *section*, *summary*, které jsou ve specifikaci HTML 5, a jde o blokové elementy, které jsou v exportovaném souboru zobrazeny shodně jako blokový element značky *div*. Dalším rozšířením je značka *progress*, která zobrazuje průběh zpracování nějaké funkce v počítači, a značka *meter*, používaná pro měření dat v daném rozsahu. Dále jsou pro formuláře podporovány značka *legend*, pro definování titulku několika políček formuláře, a značka *fieldset* pro ohraničení těchto polí formuláře. Tyto značky jsou prozatím ve fázi testování, a ne vždy jsou zobrazeny správně. Došlo k podpoře kaskádových stylů, konkrétně pro vlastnosti *min-height*, *mix-width*, *max-height* a *max-width*, které definují maximální/minimální šířku/výšku prvku. Tato podpora se týká prozatím pouze obrázků, nikoliv blokových elementů.

5.11. MPDF vs. FPDF s rozšířeními

Pro třídu FPDF bylo oficiálně vydáno 90 rozšíření, které je možné stáhnout a přidat ke třídě. Tyto skripty značně rozšiřují funkcionalitu FPDF třídy, které je ve výsledku podobná třídě MPDF. Ale i s těmito rozšířeními se svojí funkcionalitou z hlediska exportu webových aplikací nedá téměř srovnávat s MPDF třídou. Bohužel, pouze částečná podpora jazyka HTML je pro použití třídy FPDF v praktickém nasazení bez dalšího přidání kódu pro export webových aplikací omezující.

Ani jedno z těchto rozšíření pro FPDF třídu neumí exportovat tabulky s dynamickou šířkou sloupců. Respektive, před zahájením exportu je nutné znát šířku každého sloupce. Výjimkou je výška řádku, která je již automaticky dopočítávána, zatímco v původní FPDF bylo nutné znát i výšku řádku. Dokonce ani zápis tabulky pomocí jazyka HTML neumožňuje dynamické upravení šířky sloupce dle rozměrů stránky a rozměru obsahů položek sloupce. V tomto případě je nutné zadat u každého sloupce atribut *width*, který definuje šířku daného prvku. V případě definování příliš

velké šířky pro každý sloupec dojde k vypsaní mimo okraje stránky. Pro podporu a export sloupců pomocí kaskádových stylů neexistuje žádné rozšíření.

Pro podporu UTF-8 kódování by bylo nutné třídu FPDF přeprogramovat, protože pro zápis používá funkce, které UTF-8 kódování nepodporují. Přeprogramováním této třídy vznikla nová třída – tFPDF, která UTF-8 kódování podporuje. I když jde o upravení původní FPDF třídy, je zobrazena mezi rozšířeními. V rámci této třídy jsou přidávány použité znaky do exportovaného souboru a výsledný soubor je tak větší o použité znaky při exportu.

Existuje i rozšíření pro jazyk HTML, které tuto třídu pro použití při exportu webových aplikací značně vylepšuje, podpora značek je poměrně nízká. Podporovány jsou pouze nadpisy do úrovně 4, obrázky, tabulky, číslované seznamy, citace. V případě exportu tabulek nedojde k žádnému rozlišení okolních buněk. Je vypsan pouze obsah jednotlivého řádku tabulky, bez žádného formátování, pouze smazání značek pro jednotlivé sloupce řádku. Existují sice rozšíření pro export tabulek, která ale nejsou určena pro export jazyka HTML, a každé rozšíření má své omezení, např. není umožněn export rozsáhlých tabulek v rozsahu více než jedné stránky. Při exportu jazyka HTML pomocí tohoto rozšíření jsou automaticky vytvořeny záhlaví a zápatí stránky. Záhlaví obsahuje titulek, URL a autora exportované stránky, dále datum exportu. V záhlaví je vypsané pouze číslo stránky s celkovým počtem stránek.

Oproti třídě MPDF je pomocí třídy FPDF s některými rozšířeními možné zapsat do PDF souboru prvky, které třída MPDF nepodporuje, například rozšíření pro podporu exportu kalendáře USA nebo vykreslení hvězdy do exportovaného souboru, zde se jedná spíše o grafickou ukázkou možností než o praktické použití, stejně tak, jako vykreslení mnohoúhelníků do exportovaného souboru. MPDF třída je nástroj pro export primárně jazyka HTML. Zatímco některá rozšíření pro FPDF třídu umožňují export dat z databáze, jak MySQL, tak PostgreSQL.

Třída FPDF s rozšířeními je tedy s porovnáním s třídou MPDF značně nepoužitelná pro přímé použití tiskového výstupu webových aplikací. Třidu MPDF je možné použít ihned pro praktické nasazení. Zatímco třída FPDF je vhodná spíše pro naprogramování několika dalších funkcí a následné použití pro export pouze konkrétních jednoduchých webových aplikací, pro která není nutné používat rozsáhlou MPDF třídu. Všechna rozšíření pro tuto třídu jsou pod stejnou licencí jako třída sama,

tedy pod svobodnou licenci. Třída sama i všechny její rozšíření je možné upravovat dle záměru a využít pro komerční i nekomerční projekt.

Příloha 10 obsahuje tabulku, kde jsou popsána všechna rozšíření, kdy a kým byla vydána a popis jejich funkcionality.

6. Závěr

Cílem práce bylo vytvořit tiskový výstup pro webové aplikace. Tento výstup byl založen na MPDF třídě, která plně podporuje jazyk HTML, a je přímo určená pro export webových aplikací do formátu PDF. Je napsána v jazyce PHP, jedná se tedy o serverové řešení, a není závislé na uživatelské platformě. Díky její funkcionalitě a stálému vývoji slouží jako vhodné řešení pro export těchto aplikací. Prozatím částečná podpora CSS 3 a HTML 5 z ní dělá nadčasový software, protože některé z těchto funkcí prozatím nejsou podporovány dnešními internetovými prohlížeči. Tato třída byla tedy zanalyzována a porovnány její exporty stránek do formátu PDF s tiskovým výstupem prohlížečů. Dále byla zdokumentována funkcionalita, kterou třída umožňuje oproti standardnímu HTML.

Nastavení MPDF třídy umožňuje značně změnit rychlost exportu, velikost této třídy a zatížení serveru. Od tohoto nastavení se mění i velikost exportovaného souboru. Záleží na programátorovi aplikace, co vše je potřeba exportovat a s jakým výchozím nastavením. Při exportu je nutné brát ohled na zatížení serveru a dobu exportu. Tato třída je pro exportování mnohem více vhodná než například třída HTML2PDF, u které dochází k neidentifikovaným chybám. Navíc při exportu pomocí HTML2PDF dochází k rozdílnému a často i nečitelnému formátování textu, například výpis několikařádkového textu pouze do jednoho řádku.

Třída MPDF zobrazuje vyexportovaný soubor téměř totožně jako internetové prohlížeče, ale ne ve všech případech. U složitějších stránek s obsahem několika vnořených blokových prvků docházelo k pozměnění jejich rozměrů a vlastností. Pro shodné zobrazení je podmínkou, aby styl byl definován přímo v konkrétní značce pomocí atributu *style*. Při splnění této podmínky budou akceptovány všechny vlastnosti vnořeného blokového prvku.

Třída MPDF podporuje téměř všechny HTML značky a CSS vlastnosti, až na několik výjimek. Jelikož formát PDF nepodporuje HTML značky ani CSS vlastnosti, jsou hodnoty značek při exportu konvertovány a poté zapsány do PDF. Z tohoto důvodu není zobrazení exportu webové aplikace vždy shodné se zobrazením tiskového výstupu aplikace v internetovém prohlížeči. Ať už se jedná o velikost písma či okraje tabulek.

Třída MPDF akceptuje HTML komentáře, které jsou určeny pouze pro tuto třídu. Díky této možnosti je možné napsat styl, který je určen pouze pro tuto třídu MPDF k exportu webových aplikací do formátu PDF. Třída umožňuje i zápis několika značek, jejichž zobrazení není možné dosáhnout pomocí jazyka HTML, například definování záhlaví a zápatí stránek.

Přínosem práce je kompletní shrnutí a rozebrání tiskových výstupů webových aplikací, včetně exportů do formátu PDF. Byly rozebrány jejich výhody, nevýhody, omezení a podrobná analýza třídy MPDF, konkrétně porovnání vytížení serveru a časové náročnosti při exportu rozsáhlých tabulek s různým nastavením této třídy, podpora cizích fontů a především porovnání rozdílného zobrazení s tiskovým výstupem webových aplikací, především s internetovými prohlížeči Google Chrome a Mozilla Firefox, jelikož tyto prohlížeče nejsou závislé na platformě. Zatímco verze Internet Exploreru 9 není možné spustit na starších operačních systémech, konkrétně Windows XP a starší.

7. Seznam použité literatury

- [1] Chris Waterson, *How browsers work*, 10.2009, URL:
<http://taligarsiel.com/Projects/howbrowserswork1.htm>
- [2] Adobe Systems, *Adobe*[online], 2013, URL: <http://www.adobe.com>
- [3] Olivier Plathey, *FPDF*[online], 3. 11. 2012, URL: <http://www.fpdf.org>
- [4] Ian Back, *MPDF*[online], 9. 6. 2008, URL: <http://www.mpdf1.com>
- [5] PHP Group, *PHP: Hypertext Preprocessor*[online], 3. 12. 2006, URL:
<http://www.php.net>
- [6] Autrijus Tang, *HTML2PDF*[online], 5. 4. 2007,
<http://html2fpdf.sourceforge.net>
- [7] PHPMyAdmin team, *PHPMyAdmin*[online], 2013,
<http://www.phpmyadmin.net>
- [8] Dušan Janovský, *Jak psát web*[online], 31. 3. 2013,
<http://www.jakpsatweb.cz>

8. Přílohy

Příloha 1 - Ukázka exportu pomocí HTML2(F)PDF stránky www.nti.tul.cz.

Novinky

iNTEC

Vstani? OX ha belu? Nesti? Ute?uje nov? na

Dostupnost po IPv6

[illegible]

Finan?n? podporovaná témata semestrálních a diplomových prací

Abstract

Datové úložišť? Bacula

BTB dělnického výtvaru je vám k dispozici

O ústavu

bioRxiv preprint doi: <https://doi.org/10.1101/091547>; this version posted July 10, 2016. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Výzkum

Organizační struktura ústavu je dána výzkumnými týmy řešícími aktuální výzkumné projekty.

Stellen Sie sich vor, Sie sind ein Wissenschaftler, der die Auswirkungen von Umweltveränderungen auf die Artenvielfalt untersucht. Sie haben eine große Anzahl von Daten, die die Artenvielfalt in verschiedenen Ökosystemen über die Zeit hinweg zeigen. Diese Daten sind in einer Tabelle dargestellt, die die Artenvielfalt in verschiedenen Ökosystemen über die Zeit hinweg zeigt. Die Tabelle ist in Spalten unterteilt, die die Artenvielfalt in verschiedenen Ökosystemen über die Zeit hinweg zeigen. Die Spalten sind: Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung. Die Zeilen sind: Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung. Die Tabelle ist in Spalten unterteilt, die die Artenvielfalt in verschiedenen Ökosystemen über die Zeit hinweg zeigen. Die Spalten sind: Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung. Die Zeilen sind: Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung, Artenvielfalt, Ökosystem, Zeit, Umweltveränderung.

- Projekt MPO-TIP „Výzkum vlastností materiálů pro bezpečné ukládání radioaktivních odpadů“ a vývoj postupů jejich hodnocení (BPMAT) vedený doc. Milanem Hokrem je zaměřen na získání informací a dat pro budoucí stavbu hlubinného úložiště vyhořelého jaderného paliva. Cílem je vypracování metodik, nástrojů a postupů pro hodnocení a výběr materiálů z hlediska bezpečnosti úložiště. Partneři tohoto projektu, který vede ÚJV Řež a.s., jsou dále VVUT a VCHT.
- Cílem projektu MPO-TIP „Výzkum a vývoj nanomateriálů pro filtraci – snížení emisí ze spalín a přemyslových plynů“ (NANOFI) vedeného prof. Jiřím Maryškou je technologie přepravy vrstevnatých filtračních materiálů složených z textilního nosiče, nanovlákněné vrstvy a katalyzátorů. Materiál bude využit na výrobu filtračních elementů pro čištění a katalytický rozklad nežádoucích složek plynů. Dalšími partnery projektu jsou TERMIZO a.s., ELMARCO s.s.r.o. a ECOTEC.
- Výzkumný záměr „Optická síť národního výzkumu a její nové aplikace“ řešený pod vedením doc. Pavla Satrapy v rámci sdružení CESNET má za cíl vyvíjet inovativní síťové technologie a aplikace a uplatňovat je při rozvoji národní sítě pro vzdělávání.
- Přemyslový výzkum a vývoj zahrnuje firemní projekty a zakázky. Mezi nejvýznamnější patří poskytování konzultací pro firmu RWE Gas Storage, s.r.o., které vede doc. Otto Severýn a týká se interpretace seismických měření v oblasti vřtárenské předhlubně v lokalitě Lobodice-Troubky-Petrov. Dále i zakázky pro firmy AQUATEST a MEGA zahrnující laboratorní zkoušky vývoje metodik pro in-situ sanační metody (ovvěření účinnosti inidel) na vzorcích kontaminované vody a zeminy, které vede doc. Miroslav Ěrník.

Abstract The purpose of this study was to determine the effect of a 12-week training program on the physical fitness and health-related quality of life (HRQL) of sedentary, middle-aged, obese women. The study was a randomized, controlled trial. The intervention group (IG) participated in a 12-week training program, while the control group (CG) remained sedentary. The IG showed significant improvements in physical fitness and HRQL compared to the CG. The findings suggest that a 12-week training program can improve physical fitness and HRQL in sedentary, middle-aged, obese women.

Vážená, náš inštitút ako je CTRF40S rála ANVBA a ďalší aktivitami a spolupracami n?kterých ?len? NTI na projektech

Vzdělávání

Váš zápis: Zimová chemie je...? v sešitě vykašlete hodnoty i experimentální! Zimní práce: k předchozí práci přidejte v sekci

DOI: 10.1002/for

Ve spolupráci s VUT se ústav podílí na využití „Meziuniverzitní podzemní laboratoře (MEZILAB)“, která vznikla v roce 2010 v

Příloha 2 - Porovnání náhledu tabulky v prohlížeči a v exportovaném souboru.

HTML

Hlavička tabulky tag caption

Hlavičková buňka tag th	Hlavičková buňka tag th
Buňka tabulky tag td s atributem rowspan=2	Buňka tabulky tag td Buňka tabulky tag td
Buňka tabulky tag td s atributem colspan=2	Buňka tabulky tag td
Buňka tabulky tag td	Buňka tabulky tag td
Vnořená tabulka	o rozměrech 2 x 2
Buňka vnořené tabulky	Buňka vnořené tabulky
Vnořený tag div v tabulce	
Vnořený tag p v tabulce	
Buňka tabulky tag td s rodičovským tagem tfoot	

MPDF

Hlavička tabulky tag caption

Hlavičková buňka tag th	Hlavičková buňka tag th
Buňka tabulky tag td s atributem rowspan=2	Buňka tabulky tag td Buňka tabulky tag td
Buňka tabulky tag td s atributem colspan=2	Buňka tabulky tag td
Buňka tabulky tag td	Buňka tabulky tag td
Vnořená tabulka	o rozměrech 2 x 2
Buňka vnořené tabulky	Buňka vnořené tabulky
Vnořený tag div v tabulce	
Vnořený tag p v tabulce	
Buňka tabulky tag td s rodičovským tagem tfoot	

Příloha 3 - Fyzické a logické formátování textu

HTML

Odstavec s Fyzickým formátováním textu

Předformátovaný text tagem pre v celém odstavci:
Tučný text tag b Podtržený text tag u Kurzíva tag i
Horní index tag sup
Dolní index tag sub
Zmenšení textu tag small Přeskrtnutý text tag s
Zvětšení textu tag big Přeskrtnutý text tag strike
Teletype tag tt
Písmo tag font s atributy color="red" size="5" face="Arial"

Odstavec s logickým formátováním textu

Odstavec tag p se zalamováním řádku
Tučný text tag strong
Kurzíva tag em
Citace tag cite
Výpis kódu tag code
Nově použitý termín tag dfn
Ukázka tag samp
Formátování proměnné tag var
Ustálený výraz tag abbr
Zkratka tag acronym
smazaný obsah tag del
Přidaný text tag ins
"Citace tag q"

MPDF

Odstavec s Fyzickým formátováním textu

Předformátovaný text tagem pre v celém odstavci:
Tučný text tag b Podtržený text tag u Kurzíva tag i
Horní index tag sup
Dolní index tag sub
Zmenšení textu tag small Přeskrtnutý text tag s
Zvětšení textu tag big Přeskrtnutý text tag strike
Teletype tag tt
Písmo tag font s atributy color="red" size="5" face="Arial"

Odstavec s logickým formátováním textu

Odstavec tag p se zalamováním řádku
Tučný text tag strong
Kurzíva tag em
Citace tag cite
Výpis kódu tag code
Nově použitý termín tag dfn
Ukázka tag samp
Formátování proměnné tag var
Ustálený výraz tag abbr
Zkratka tag acronym
smazaný obsah tag del
Přidaný text tag ins
Citace tag q

Příloha 4 - Export tabulek s různě definovanou hodnotou rámečku vlastností border-style

HTML	MPDF
Buňka tabulky s border-style:solid	Buňka tabulky s border-style:solid
Buňka tabulky s border-style:dotted	Buňka tabulky s border-style:dotted
Buňka tabulky s border-style:dashed	Buňka tabulky s border-style:dashed
Buňka tabulky s border-style:groove	Buňka tabulky s border-style:groove
Buňka tabulky s border-style:ridge	Buňka tabulky s border-style:ridge
Buňka tabulky s border-style:double	Buňka tabulky s border-style:double
Tabulka s border-collapse: separate	Tabulka s border-collapse: separate

Příloha 5 - V levé části náhled z exportu MPDF, v pravé části náhled v prohlížeči Mozilla Firefox. Export stránky www.nti.tul.cz

Hledat



- Navigace
 - Hlavní strana
 - Členové ústavu
 - Výuka
 - Věda a výzkum
 - Seminář NTI
 - Služby
 - Nápořádě

- Osobní nástroje
 - Přihlaste se

Z Ústav nových technologií a aplikované informatiky

O ústavu

Ústav Nových technologií a aplikované informatiky vznikl při reorganizaci Fakulty mechatroniky, informatiky a mezinárodních studií 1. ledna 2007 integrací kateder a výzkumného centra Pokročilé sanační technologie a procesy. Multidisciplinární kolektiv ústavu připravuje a řeší výzkumné projekty charakterizované průmyslové uplatněním výstupu. Výzkumná a vzdělávací činnost NTI je zaměřena na výzkum a aplikace počítačových modelů, moderních informačních technologií a nanotechnologií pro studium přírodních a

Novinky

INTEC

Větší část našeho ústavu pracuje nově na ústavu CXI v oddělení INTEC.

Dostupnost po IPv6

technických procesů za účelem jejich ovlivňování a řízení.

Výzkum

Organizační struktura ústavu je dána výzkumnými týmy řešícími aktuální výzkumné projekty.

V současnosti jsou hlavní výzkumné kapacity ústavu soustředěny ve Výzkumném centru Pokročilé sanační technologie a procesy (ARTEC) vedeném prof. Jiřím Maryškou s navazující širokou spoluprací s dalšími ústavy fakulty (především MTI, pracovníci TUL, ale i pracovními skupinami z jiných institucí (CGS, AQUATEST). Výzkumná činnost centra je zaměřena na studium přírodních procesů v biosféře a vlivů cílených zásahů do tohoto prostředí, tvorbu nových metodik a nástrojů pro řízení těchto procesů a jejich ekonomického hodnocení.

V oblasti aplikovaného výzkumu jsou kapacity ústavu zaměřeny na řešení výzkumných projektů a na přímou spolupráci s podniky.

- Projekt MPO-TIP „Výzkum vlastností materiálů pro bezpečné ukládání radioaktivních odpadů a vývoj postupů jejich hodnocení“ (BPMAT) vedený doc. Milanem Hokrem je zaměřen na získání informací a dat pro budoucí stavbu hlubinného úložiště vyhořelého jaderného paliva. Cílem je vypracování metodik, nástrojů a postupů pro hodnocení a výběr materiálů z hlediska bezpečnosti úložiště. Partneři tohoto projektu, který vede ÚJV Řež a.s., jsou dále ČVUT a VŠCHT.
- Cílem projektu MPO-TIP „Výzkum a vývoj nanomateriálů pro filtraci - snížení emisí ze spalín a průmyslových plynů“ (NANOFIL) vedeného prof. Jiřím Maryškou je technologie přípravy vrstevnatých filtračních materiálů složených z textilního nosiče, nanovláknenné vrstvy a katalyzátorů. Materiál bude využit na výrobu filtračních elementů pro čištění a katalytický rozklad nežádoucích složek plynů. Dalšími partnery projektu jsou TERMIZO a.s., ELMARCO s.s.r.o. a ECOTEX.
- Výzkumný záměr „Optická síť národního výzkumu a její nové aplikace“ řešený pod vedením doc. Pavla Satrapy v rámci sdružení CESNET má za cíl vyvíjet inovativní síťové technologie a aplikace a uplatňovat je při rozvoji národní sítě pro vědu, výzkum a vzdělávání.
- Průmyslový výzkum a vývoj zahrnuje firemní projekty a zakázky. Mezi nejvýznamnější patří poskytování konzultací pro firmu RWE Gas Storage, s.r.o., které vede doc. Otto Severýn a týká se interpretace seismických měření v oblasti vřekarpatské předhlubně v lokalitě Lobodice-Troubky-Přerov. Dále i zakázky pro firmu AQUATEST a MEGA zahrnující laboratorní zkoušky a vývoj metodik pro in-situ sanační metody (ověření účinnosti činidel) na vzorcích kontaminované vody a zeminy, které vede doc. Miroslav Černík.

Mezinárodní spolupráce představují jednak projekt 7. Rámcového programu Vývoj konceptu pro intenzifikaci čištění vod za pomoci integrovaných nano a membránových technologií, jehož podstatou je zlepšit fyzikální a chemické vlastnosti polymerních membrán a tím zlepšit jejich filtrační vlastnosti a vede ho doc. Miroslav Černík, dále projekt DECOVALEX-2011 sdružující pracovníky z 8 zemí Evropy a Asie zaměřený na výzkum T-H-M-C procesů v souvislosti s hlubinným ukládáním vyhořelého jaderného paliva vedený doc. Milanem Hokrem a mezinárodní projekt „Mobile Systems SW Engineering Education Network“ (MobEduNet) vedený doc. Pavlem Satrapou.

Výzkumná činnost ústavu je dále doplněna individuálními aktivitami a spoluprací některých členů NTI na projektech partnerských institucí jako je CERFACS, UT AV CR a další.

Hledat



- Navigace
 - Hlavní strana
 - Členové ústavu
 - Výuka
 - Věda a výzkum
 - Seminář NTI
 - Služby
 - Nápořádě

- Osobní nástroje
 - Přihlaste se

Z Ústav nových technologií a aplikované informatiky

O ústavu

Ústav Nových technologií a aplikované informatiky vznikl při reorganizaci Fakulty mechatroniky, informatiky a mezinárodních studií 1. ledna 2007 integrací kateder a výzkumného centra Pokročilé sanační technologie a procesy. Multidisciplinární kolektiv ústavu připravuje a řeší výzkumné projekty charakterizované průmyslové uplatněním výstupu. Výzkumná a vzdělávací činnost NTI je zaměřena na výzkum a aplikace počítačových modelů, moderních informačních technologií a nanotechnologií pro studium

Novinky

INTEC

Větší část našeho ústavu pracuje nově na ústavu CXI v oddělení INTEC.

přírodních a technických procesů za účelem jejich ovlivňování a řízení.

Výzkum

Organizační struktura ústavu je dána výzkumnými týmy řešícími aktuální výzkumné projekty.

V současnosti jsou hlavní výzkumné kapacity ústavu soustředěny ve Výzkumném centru Pokročilé sanační technologie a procesy (ARTEC) vedeném prof. Jiřím Maryškou s navazující širokou spoluprací s dalšími ústavy fakulty (především MTI, pracovníci TUL, ale i pracovními skupinami z jiných institucí (CGS, AQUATEST). Výzkumná činnost centra je zaměřena na studium přírodních procesů v biosféře a vlivů cílených zásahů do tohoto prostředí, tvorbu nových metodik a nástrojů pro řízení těchto procesů a jejich ekonomického hodnocení.

V oblasti aplikovaného výzkumu jsou kapacity ústavu zaměřeny na řešení výzkumných projektů a na přímou spolupráci s podniky.

- Projekt MPO-TIP „Výzkum vlastností materiálů pro bezpečné ukládání radioaktivních odpadů a vývoj postupů jejich hodnocení“ (BPMAT) vedený doc. Milanem Hokrem je zaměřen na získání informací a dat pro budoucí stavbu hlubinného úložiště vyhořelého jaderného paliva. Cílem je vypracování metodik, nástrojů a postupů pro hodnocení a výběr materiálů z hlediska bezpečnosti úložiště. Partneři tohoto projektu, který vede ÚJV Řež a.s., jsou dále ČVUT a VŠCHT.
- Cílem projektu MPO-TIP „Výzkum a vývoj nanomateriálů pro filtraci - snížení emisí ze spalín a průmyslových plynů“ (NANOFIL) vedeného prof. Jiřím Maryškou je technologie přípravy vrstevnatých filtračních materiálů složených z textilního nosiče, nanovláknenné vrstvy a katalyzátorů. Materiál bude využit na výrobu filtračních elementů pro čištění a katalytický rozklad nežádoucích složek plynů. Dalšími partnery projektu jsou TERMIZO a.s., ELMARCO s.s.r.o. a ECOTEX.
- Výzkumný záměr „Optická síť národního výzkumu a její nové aplikace“ řešený pod vedením doc. Pavla Satrapy v rámci sdružení CESNET má za cíl vyvíjet inovativní síťové technologie a aplikace a uplatňovat je při rozvoji národní sítě pro vědu, výzkum a vzdělávání.
- Průmyslový výzkum a vývoj zahrnuje firemní projekty a zakázky. Mezi nejvýznamnější patří poskytování konzultací pro firmu RWE Gas Storage, s.r.o., které vede doc. Otto Severýn a týká se interpretace seismických měření v oblasti vřekarpatské předhlubně v lokalitě Lobodice-Troubky-Přerov. Dále i zakázky pro firmu AQUATEST a MEGA zahrnující laboratorní zkoušky a vývoj metodik pro in-situ sanační metody (ověření účinnosti činidel) na vzorcích kontaminované vody a zeminy, které vede doc. Miroslav Černík.

Mezinárodní spolupráce představují jednak projekt 7. Rámcového programu Vývoj konceptu pro intenzifikaci čištění vod za pomoci integrovaných nano a membránových technologií, jehož podstatou je zlepšit fyzikální a chemické vlastnosti polymerních membrán a tím zlepšit jejich filtrační vlastnosti a vede ho doc. Miroslav Černík, dále projekt DECOVALEX-2011 sdružující pracovníky z 8 zemí Evropy a Asie zaměřený na výzkum T-H-M-C procesů v souvislosti s hlubinným ukládáním vyhořelého jaderného paliva vedený doc. Milanem Hokrem a mezinárodní projekt „Mobile Systems SW Engineering Education Network“ (MobEduNet) vedený doc.

Vzdělávání

Vzdělávací činnost zahrnuje předměty strukturovaného studijního programu zaměřené na numerické metody, stavbu počítačových modelů, programování a webových technologií a experimentálních technik a postupů. Podrobnosti najdete v sekci Výuka.

Dále je ústav zapojen do řady vzdělávacích projektů především v rámci programu OP VK. Mezi nejvýznamnější patří projekty „MultiEdu“ zaměřené na streamování přednášek a jejich šíření prostřednictvím webu, který vede dr. Klára Čisarová (MTI). Projekty „Vyhledávání talentů“ a „Zvýšení zájmu SŠ studentů o studium technických oborů“ jsou zaměřeny na oslovení nadaných studentů SŠ pro zvýšení jejich zájmu o studium technických a technologických oborů. Projekty vedou doc. Miroslav Černík a Ing. Miloš Hernych (MTI).

Ve spolupráci s ČVUT se ústav podílí na využití „Meziuniverzitní podzemní laboratoře (MEZILAB)“, která vznikla v roce 2010 v rámci rozvojového projektu MSM, zaměřeného na rozšíření výuky o cvičení v terénu, ve štolě Josef spravované jako výzkumné a výukové středisko ČVUT. Vedoucím na TUL je doc. Milan Hokr

Citováno z „http://www.nti.tul.cz/cz/Hlavni%C3%AD_strana“

Stránka byla naposledy editována 18. 3. 2013 v 12:44. Stránka byla zobrazena 118

152krát.

• Design: Jiří Hnídek, 2009-2012
• O webu NTI



Pavlem Satrapou.

Výzkumná činnost ústavu je dále doplněna individuálními aktivitami a spoluprací některých členů NTI na projektech partnerských institucí jako je CERFACS, UI AV ČR a další.

Vzdělávání

Vzdělávací činnost zahrnuje předměty strukturovaného studijního programu zaměřené na numerické metody, stavbu počítačových modelů, programování a webových technologií a experimentálních technik a postupů. Podrobnosti najdete v sekci Výuka.

Dále je ústav zapojen do řady vzdělávacích projektů především v rámci programu OP VK. Mezi nejvýznamnější patří projekty „MultiEdu“ zaměřené na streamování přednášek a jejich šíření prostřednictvím webu, který vede dr. Klára Čisarová (MTI). Projekty „Vyhledávání talentů“ a „Zvýšení zájmu SŠ studentů o studium technických oborů“ jsou zaměřeny na oslovení nadaných studentů SŠ pro zvýšení jejich zájmu o studium technických a technologických oborů. Projekty vedou doc. Miroslav Černík a Ing. Miloš Hernych (MTI).

Ve spolupráci s ČVUT se ústav podílí na využití „Meziuniverzitní podzemní laboratoře (MEZILAB)“, která vznikla v roce 2010 v rámci rozvojového projektu MSM, zaměřeného na rozšíření výuky o cvičení v terénu, ve štolě Josef spravované jako výzkumné a výukové středisko ČVUT. Vedoucím na TUL je doc. Milan Hokr

Citováno z „http://www.nti.tul.cz/cz/Hlavni%C3%AD_strana“

Stránka byla naposledy editována 18. 3. 2013 v 12:44. Stránka byla zobrazena 118 181krát.

Design: Jiří Hnídek, 2009-2012
O webu NTI



Příloha 6 - Export MPDF a kaskádové styly verze 3 – stín pod textem a pod rámečkem, zakulacení rohů rámečku. Záhloví a zápatí pomocí značek pro MPDF a 3 sloupce.

Testování hlavičky

malesuada malesuada Vivamus pretium eleifend Nunc. Adipiscing in enim eget orci vitae accumsan massa consequat interdum scelerisque. Magna.

Lorem ipsum dolor sit amet consectetur egestas arcu vitae ut urna. Nulla ipsum ligula orci nibh ut Maecenas sodales ut.

Sagittis eget neque magnis volutpat commodo Pellentesque laoreet libero convallis Aenean. Purus laoreet vitae sapien Curabitur ut sed et ipsum porttitor

~~Lorem ipsum dolor sit amet consectetur egestas arcu vitae ut urna. Nulla ipsum ligula orci nibh ut Maecenas sodales ut.~~

Sagittis eget neque magnis volutpat commodo Pellentesque laoreet libero convallis Aenean. Purus laoreet vitae sapien Curabitur ut sed et ipsum porttitor dui. Non vitae eget Vestibulum Nulla convallis Integer pretium at vitae dolor. Quis.

Urna Maecenas fringilla congue quis malesuada malesuada Vivamus pretium eleifend Nunc. Adipiscing in enim eget orci vitae accumsan massa consequat interdum scelerisque. Magna.

dui. Non vitae eget Vestibulum Nulla convallis Integer pretium at vitae dolor. Quis.

Urna Maecenas fringilla congue quis malesuada malesuada Vivamus pretium eleifend Nunc. Adipiscing in enim eget orci vitae accumsan massa consequat interdum scelerisque. Magna.

Lorem ipsum dolor sit amet consectetur egestas arcu vitae ut urna. Nulla ipsum ligula orci nibh ut Maecenas sodales ut.

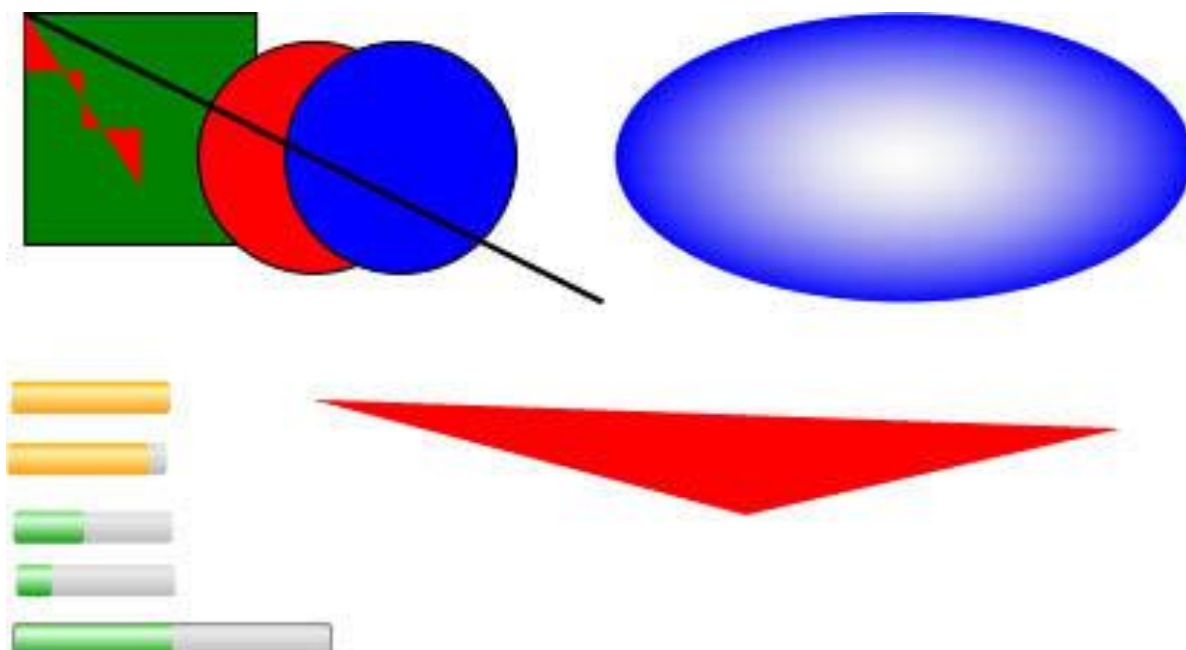
Sagittis eget neque magnis volutpat commodo Pellentesque laoreet libero convallis Aenean. Purus laoreet vitae sapien Curabitur ut sed et ipsum porttitor dui. Non vitae eget Vestibulum Nulla convallis Integer pretium at vitae dolor. Quis.

Urna Maecenas fringilla congue quis malesuada malesuada Vivamus pretium eleifend Nunc. Adipiscing in enim eget orci vitae accumsan massa consequat interdum scelerisque. Magna.

1

10-04-2013

Příloha 7 - HTML 5 ve třídě MPDF a vektorová grafika



Příloha 8 - Formuláře a aktivní formuláře v MPDF třídě

Formuláře

tag input ty

tag label pro následující input typu password

tag radio ☐

tag checkbox ☒

tag select ▼

tag input ty

tag textarea s
šířkou(cols) 20 znaků a
výškou(rows) 3 řádky

tag fieldset pro 2 inputy

tag legend

tag input ty

tag button

tag input typu submit

Aktivní formuláře

Text tag input typu

tag label pro následující input typu password

□□□□

File

tag radio ☒

✗

tag checkbox ☒

Select

tag select ▼

tag input typu

tag textarea s šířkou
(cols) 20 znaků a výškou
(rows) 3 řádky

tag fieldset pro 2 inputy

tag legend

tag input typu

tag button

tag input typu submit

reset hodnot

Příloha 9 - Export vnořených blokových elementů, pravý horní obrázek je náhled prohlížeče Google Chrome, vše ostatní jsou jednotlivé stránky exportu.

- Fakulta strojní
- Fakulta textilní
- Fakulta přírodovědně-humanitní a pedagogická
- Ústav pro nanomateriály, pokročilé technologie a inovace
- Ekonomická fakulta
- Ústav zdravotnických studií
- Fakulta mechaniky, inženýrské a architektury

- Studenti
 - Legislativa
 - Studentická grantová soutěž
 - Výuka
 - Stipendia
 - Programy výměny studentů
 - Učebnice ceny
 - Identifikační průkazy
 - Koleje a menzy
 - Sport

- Stabilita zaměstnání
- Studentická komora AS TUL
- Zaměstnanci
 - Legislativa
 - Zaměstnání
 - Zahraniční sítě
 - Organizační struktura
 - Sport
 - Projekty na TUL
 - Ostatní
- Propagace TUL
- Zájemci o studium
 - Univerzita
 - Studijní programy
 - Příjmací řízení
 - Koleje a ubytování
 - Sport
 - www.jetulip.cz
 - O univerzitě
- Veřejnost
 - Telefonní seznam
 - Organizační struktura
 - Mezinárodní spolupráce
 - Věda a výzkum
 - Projekty FRUŠ
 - Ostatní projekty
 - Informace pro veřejnost
 - Projekty na TUL
 - O univerzitě
- Úřední deska
 - Úřední deska TUL
 - Úřední desky fakult a ústavů
 - Veřejné zakázky

Novinky z prostředí TUL

Hudba, hudba, divadlo, píti a snad i slunce
O ložiskům majáků dosahy teploty až 35 stupňů Celsia. Letos zůstane tuť v teplotních azí více při zemi, ale hudby bude opět dost.
18.04. 2013 | celý článek

Studenti připravují studii rozvoje města Aše
Studenti naší fakulty umění a architektury chtějí navrhnout zásahy do architektury města Aše a ohledem na připravované investice územních developerů.
18.04. 2013 | celý článek

Výzva z Liberce: darujeme kostní dířek
Výzvu ke vstupu do registru dárců krve zveřejnila dnes Studentická unie naší univerzity. A padla na úrodnou půdu.
09.04. 2013 | celý článek

TUL nabízí bezplatný letní dětský tábor
Čtrnáct dnů bádní v univerzitních laboratorních proleženého sportem a pobytém v přírodě nabízí dětem ve věku 10 až 15 let naše univerzita
03.04. 2013 | celý článek

28.04. 2013 | celý článek

Výzva studentů „Pomozme nemocným leukémiím“
V boj se zákeřnou nemoc chtějí své kolegy Vědy pomoci studenti naší univerzity. Vyzývají kolegy ke vstupu do Českého registru dárců krve tvorných buněk.
05.04. 2013 | celý článek

Mladí vědci mají otevřenou cestu do Japonska
Hiroyasu Kana, docent japonské Shingu University, navštívil naši fakultu textilu. S jej díkankou hovořil o výměně mladých vědců.
03.04. 2013 | celý článek

Dana Drábková nabádá: studujte techniku!
Do aplikací moderních technologií v energetickém průmyslu se musí zapojit mladá generace. Měla by více studovat techniku.
03.04. 2013 | celý článek

ECTS / DS lákají
Kde nás najdete? Elektronická přihláška IS-STAG Elektronická přihláška jetulip.cz Absolvování tui Tetra Facebook LANE

REKTORÁT
FAKULTA STROJNÍ
FAKULTA TEXTILNÍ
FAKULTA PŘÍRODOVĚDNĚ-HUMANITNÍ A PEDAGOGICKÁ
EKONOMICKÁ FAKULTA
FAKULTA UMĚNÍ A ARCHITECTURY
FAKULTA MECHANIKY, INŽENÝRSTVÍ A MEZODROBNÝCH STUDIÍ
FAKULTA ZDRAVOTNICKÝCH STUDIÍ
CENTRUM DALŠÍHO VZDĚLÁVÁNÍ
ÚSTAV PRO NANOMATERIÁLY, POKROČILÉ TECHNOLOGIE A INOVACE
UNIVERZITA NISA
G-KOMPLEX

60. výročí univerzity
Kalendář akcí
Telefonní seznam
Univerzitní knihovna
Koleje a menzy
Univerzitní sport
Objednávky do menzy
Dny otevřených dveří

Akademická poradna a centrum podpory
Akademická koordinátní středisko v Euroregionu LANE

Příloha 10 - Volitelné možnosti třídy MPDF

Volba	Velikost souboru	Využití paměti RAM při vytvoření instance třídy	Význam
Žádná volitelná vlastnost	320kB	2.25MB	Bez podpory žádné z volitelných možností
HTML+CSS	565kB	3.25MB	Podpora pro základní HTML a CSS
Directw	322kB	2.25MB	Funkce pro přímý zápis do PDF souboru
Tables	320kB	2.25MB	Podpora pro tabulky psané v HTML
Lists	320kB	2.25MB	Podpora pro seznamy v HTML
Images Core	320kB	2.25MB	Podpora pro pozadí prvků a vodoznaky, obrázky JPG, PNG a
Images BMP	320kB	2.50MB	Podpora pro BMP obrázky
Images WMF	321kB	2.25MB	Podpora pro WMF obrázky
TABLES-ADVANCED-BORDER	320kB	2.25MB	Rozšířené hodnoty pro rámečky
HTMLHEADERS-FOOTERS	320kB	2.25MB	Záhlaví a zápatí pomocí HTML značek pro MPDF třídu
COLUMNS	343kB	2.25MB	Sloupce pomocí HTML značek pro MPDF třídu
TOC	330kB	2.25MB	Zarovnání na střed pomocí HTML značek pro MPDF třídu
INDEX	328kB	2.25MB	Indexování textu pomocí HTML značek pro MPDF třídu
BOOKMARKS	323kB	2.25MB	Generování obsahu pomocí HTML značek pro MPDF třídu
BARCODES	334kB	2.25MB	Generování čárových kódů pomocí HTML značek pro MPDF třídu
FORMS	320kB	2.25MB	Podpora pro aktivní formuláře
WATERMARKS	325kB	2.25MB	Podpora pro vodoznaky
CJK - FONTS	327kB	2.25MB	Podpora pro asijské CJK fonty
RTL	349kB	2.25MB	Podpora pro psaní textu zprava do leva
INDIC	323kB	2.25MB	Podpora pro indické fonty
ANNOTATIONS	328kB	2.25MB	Podpora pro generování anotací
BACKGROUNDS	320kB	2.25MB	Podpora pro vykreslování pozadí
CSS-FLOAT	320kB	2.25MB	Podpora vlastnosti <i>Float</i> pro obtékání blokových prvků
CSS-IMAGE-FLOAT	320kB	2.25MB	Podpora vlastnosti <i>Float</i> pro obrázky
CSS-POSITION	320kB	2.25MB	Podpora vlastnosti <i>Position</i>
CSS-PAGE	320kB	2.25MB	Podpora pro číslování stránek exportovaného souboru
BORDER-RADIUS	320kB	2.25MB	Podpora pro zakulacení rámečku okolo blokových elementů
HYPHENATION	326KB	2.25MB	Podpora pro dělení slov při zalamování konce řádku
ENCRYPTION	326KB	2.25MB	Zabezpečení exportovaného souboru heslem
IMPORTS	336KB	2.50MB	Importování externích souborů
PROGRESS-BAR	326KB	2.25MB	HTML5 - progress bar

Příloha 11 - Rozšíření pro FPDF třídu

Datum vydání	Název	Popis
23.2.2009	Kalendář	Export kalendáře pro USA
24.1.2009	Přímé barvy	Definování přímých barev používaných při tisku
28.6.2008	Čárový kód Code 128	Čárkové kódy
28.12.2006	Thajská podpora	Podpora pro Thajské znaky a fonty
29.10.2006	Oboustranné stránky	Formát pro sudé a liché stránky
3.9.2006	Codabar čárové kódy	Čárkové kódy
18.6.2006	podpora EPS / AI	Export vektorových obrázků typu EPS nebo AI
28.5.2006	Alfa kanál	Export PNG souborů s alfa kanálem
9.4.2006	Kruhový text	Text exportovaný do kruhu
2.10.2005	Hodiny	Export analogových hodin
4.9.2005	Proměny	Pro transformace jednotlivých prvků - rotace, posunutí, zrcadlení
15.8.2005	Ořezávání	Výřez pomocí obdélníku, textu, elipsa, kruh, mnohoúhelník.
3.7.2005	Malé velké písmena	Export textu pouze ve velkých, nebo malých písmech
12.6.2005	Optimalizace paměti	Ukládá vytvořené stránky ihned do souboru
21.5.2005	Zviditelnění	Export prvků, které jsou viditelné pouze v Adobe Readeru, ne při tisku
3.4.2005	Průhlednost	Nastavení průhlednosti jednotlivých prvků
28.3.2005	Obsah 2	Generování obsahu dokumentu
13.3.2005	Přechody	Přechod barev s lineárním a kruhovým gradientem
20.2.2005	Hvězda	Vykreslení hvězdy
13.2.2005	HTML tabulky	Rozšíření pro podporu tabulek v HTML
16.1.2005	Geometrické obrazce	Kreslí základní grafické prvky - přímka, čtverec, elipsa, atd.
24.12.2004	Rozšíření čárového kódu Code 39	Čárkové kódy
5.12.2004	Čárový kód i25	Čárkové kódy
7.11.2004	Kostka	Export 3D kostky s možností rotace ve všech osách
1.11.2004	Tekoucí blok	Textový blok
17.10.2004	Ozdobný řádek	Ohraničení, zarovnání, styly a maximální délka pro vypsání řádek
3.10.2004	Vyplnění buňky textem	Je akceptován rozměr buňky, text nepřetéká
26.9.2004	subWrite	Metoda pro formátování výpisu textu
19.9.2004	Mnohoúhelníky	Vykreslení mnohoúhelníku
12.9.2004	Načtení vlastností PDF	Pouze pro soubory vytvořené pomocí FPDF. Z exportovaného souboru vezme vlastnosti (autor, datum exportu, atd.).
5.9.2004	HTML barvy	Převod barev zadaných v HTML do hexadecimálního formátu
29.8.2004	Seskupení stránek	Číslování uvnitř skupiny stránek, umožňuje číslování pozpátku.
15.8.2004	MultiCell s odřádkama	Zápis seznamu pouze pomocí volání jedné funkce
1.8.2004	Podpora WMF	Podpora souborů formátu WMF

10.7.2004	HTML konverze	Export HTML značek
6.7.2004	TextBox	Export textového pole do ohraničeného obdélníku
20.6.2004	Nastavení zobrazení	Možnost skrytí menu, zobrazení souboru na celou obrazovku,
13.6.2004	Zápis HTML tabulky	Export HTML tabulky, nutné definovat rozměry sloupců
29.5.2004	Zalomení textu	Vynucené zalamování řádků na určité pozici
25.4.2004	Příklad s PHPlot	Vložení grafů, potřeba třída PHPlot
18.4.2004	Stromová struktura	Vykreslení stromové struktury
12.4.2004	Čárový kód Code 39	Čárkové kódy
5.4.2004	Import obrázků z paměti	Export obrázků uložených v paměti
28.3.2004	Podpora barev CMYK	Podpora pro CMYK barvy
14.3.2004	Nástroj pro SVG šablony	Nástroj na vytváření šablon pro exportované dokumenty.
1.2.2004	Rozšířený zápis HTML 2	Podpora 13 HTML značek
24.1.2004	Rozšířený zápis HTML	Podpora 5 HTML značek
22.11.2003	MultiCell s odsazením	Odsazení pro první řádek exportovaného textu
11.11.2003	MultiCell s odrážkou	Odsazení pro první řádek exportovaného textu pomocí definované odrážky
2.11.2003	Ochrana	Exportovaný soubor není možné editovat, kopírovat nebo vytisknout.
30.9.2003	Podpora JavaScript	Podpora pro Adobe JavaScript spuštěný po otevření dokumentu
27.9.2003	Zaoblený obdelník 2	Vykreslení obdélníku se zaoblenými rohy
14.9.2003	Pomlčky	Vykreslení přerušovaných čar nebo obdélníků
13.9.2003	Rozšířené funkce buňky	Export textu vodorovně, nebo svisle
28.8.2003	Rotace textu	Vykreslí rotovaný text dle nastaveného úhlu
20.7.2003	Příklad s MySQL 2	Export dat z databáze
19.7.2003	Etikety	Určené pro tisk etiket v daném formátu
14.6.2003	Diagramy	Generování koláčových grafů
17.5.2003	Čárový kód POSTNET	Čárkové kódy
10.5.2003	Přidání fontu	Přidání fontu
1.5.2003	Značky pro buňky a zvýraznění	Zvýraznění textu
21.4.2003	Dopisní zarovnání	Rozšiřuje mezery v textu dle šířky stránky
12.4.2003	Základní formátovací značky	Definování stylu pro jednotlivé prvky textu
29.3.2003	Příklad s MS Access	Exportování dat z MS Access
23.3.2003	MySQL výstup	Export dat z databáze MySQL
15.5.2003	Faktura	Ukázka pro generování faktury
8.3.2003	Kruhová výseč	Vystřihnutí obsahu kruhu
1.3.2003	Optimalizace	Normalizuje paměť při použití FPDF třídy

	paměti	
22.2.2003	Rejstřík	Export dat s indexy v textu
31.1.2003	MultiCell s maximem řádků	Exportovaný text s maximálním možným výpisem
10.1.2003	Čerchovaný obdelník	Vykreslení přerušovaného obdélníku
25.12.2002	Tabulky s MySQL	Export dat z tabulky
13.12.2002	Obsah	Generování obsahu dokumentu
11.12.2002	Tabulka se sloupcem na více stran	Rozšíření funkce pro generování tabulky
4.12.2002	Příklad s PostgreSQL	Export dat z databáze PostgreSQL
23.11.2002	Příklad s MySQL	Export dat z databáze MySQL
23.11.2002	Vodoznak	Vykreslení vodoznaku na každou použitou stránku
21.11.2002	Vynucené zarovnání	Vynucené zarovnání textu v buňkách
19.11.2002	Zaoblený obdelník	Obdélník se zakulacenými rohy
17.11.2002	Kruhy a elipsy	Vykreslení kruhu, nebo elipsy
17.11.2002	Čárový kód EAN	Čárkové kódy
17.11.2002	Výpis fontu	Umožňuje vypsat všechny znaky ve fontu Arial, Symbol a ZapfDingbats
17.11.2002	Tabulka s MultiCells	Export pro tabulky se zalomením řádku
17.11.2002	Rotace	Rotace dle středu prvku
17.11.2002	Záložky	Záložky

Příloha 11 - Obsah přiloženého CD

Hlavní adresář CD:

/fpdf/ - třída FPDF s rozšířeními

/html2(f)pdf/ - třída html2(f)pdf

/examples_html2(f)pdf/ - příklady exportu pomocí html2(f)pdf třídy

/mpdf/ - třída MPDF s rozšířeními pro podporu grafů a QR kódů

/mpdf/support/ - přehled podporovaných HTML značek a vlastností
kaskádových stylů

/examples_MPDF/ - příklady exportu pomocí MPDF třídy

/bakalarska_prace/ - bakalářská práce